



APLICACIÓN WEB PARA EL ANALISIS Y MONITOREO DE LLAMADAS ENTRANTES EN UNA EMPRESA DE CONTACT CENTER

JULIAN CAMILO SANCHEZ

**UNIVERSIDAD CATOLICA DE COLOMBIA
FACULTAD DE INGENIERIA
PROGRAMA DE INGENIERÍA DE SISTEMAS
BOGOTA D.C.
2013**

**APLICACIÓN WEB PARA EL ANALISIS Y MONITOREO DE LLAMADAS
ENTRANTES EN UNA EMPRESA DE CONTACT CENTER**

JULIAN CAMILO SANCHEZ DIAZ

Practica Empresarial

**Director
JORGE CARRILLO
Ingeniero**

**UNIVERSIDAD CATOLICA DE COLOMBIA
FACULTAD DE INGENIERIA
PROGRAMA DE INGENIERÍA DE SISTEMAS
BOGOTA D.C.
2013**



Atribución-NoComercial-SinDerivadas 2.5 Colombia (CC BY-NC-ND 2.5)

La presente obra está bajo una licencia:

Atribución-NoComercial-SinDerivadas 2.5 Colombia (CC BY-NC-ND 2.5)

Para leer el texto completo de la licencia, visita:

<http://creativecommons.org/licenses/by-nc-nd/2.5/co/>

Usted es libre de:



Compartir - copiar, distribuir, ejecutar y comunicar públicamente la obra

Bajo las condiciones siguientes:



Atribución — Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciante (pero no de una manera que sugiera que tiene su apoyo o que apoyan el uso que hace de su obra).



No Comercial — No puede utilizar esta obra para fines comerciales.



Sin Obras Derivadas — No se puede alterar, transformar o generar una obra derivada a partir de esta obra.

Nota de aceptación

Aprobado por el comité de grado en cumplimiento de los requisitos exigidos por la Facultad de Ingeniería y la Universidad Católica de Colombia para optar al título de ingenieros de Sistemas.

Ingeniero Jorge E. Carrillo C.
Director

Ingeniero Ramón Díaz
Revisor Metodológico

Bogotá D. C., Mayo 28 de 2013

A Dios por estar siempre a mi lado y permitirme terminar mis estudios
A mi madre por la paciencia y dedicación durante todo este tiempo
A mi familia por apoyarme en este proceso

AGRADECIMIENTOS

El autor expresa sus agradecimientos:

La dirección de un proyecto de grado es una tarea ardua que ha sido llevada a cabo por Jorge Carrillo, mi agradecimiento por su apoyo y seguimiento.

Al Coordinador del área de Plataformas de Contact Center Jhulian Bustos por darme la idea de desarrollar este proyecto.

Al Ingeniero de Sistemas Roger Guzmán, por sus importantes enseñanzas y aportes.

A la Ingeniera Jenny Torres por su atención y sus consejos.

A mis compañeros que me han apoyado.

CONTENIDO

	Pág.
1 INTRODUCCIÓN	15
2. PLANTEAMIENTO DEL PROBLEMA	16
2.1 RESULTADOS	17
2.2 JUSTIFICACION	18
3. OBJETIVOS DEL PROYECTO	19
3.1. OBJETIVO GENERAL	19
3.2. OBJETIVOS ESPECIFICOS	19
4. MARCO REFERENCIAL	20
4.1. MARCO CONCEPTUAL	20
4.2. MARCO TEORICO	21
4.2.1 Bases de datos	22
4.2.3 Tecnologías de información y comunicaciones (Tic)	22
4.2.4 Software libre	22
4.2.5 Redes	23
4.2.6 Viabilidad	23
5. METODOLOGIA PROPUESTA	24
6. DESARROLLO DEL PROYECTO	26
6.1. RECURSOS	26
6.2 IMPLEMENTACIÓN Y PRUEBAS	27
6.2.1 JsF	27
6.3 PRUEBAS	28
7. CONCLUSIONES	70
8. RECOMENDACIONES	71
BIBLIOGRAFIA	72

LISTA DE TABLAS

	Pág.
Tabla 1. Software utilizado en la implementación	23
Tabla 2. Glosario de términos	35
Tabla 3. Usuarios del sistema	36
Tabla 4. Descripción de tablas modelo relacional	54
Tabla 4. Requerimiento funcional registrar usuario	41
Tabla 5. Requerimiento funcional validar usuario	42
Tabla 6. Requerimiento funcional generar listado de compañías	43
Tabla 7. Requerimiento funcional generar listado de proyectos	44
Tabla 8. Requerimiento funcional generar listado de DNIS	45
Tabla 9. Requerimiento funcional generar listado de resources	46
Tabla 10. Requerimiento funcional calcular numero de llamadas entrantes	47
Tabla 11. Requerimiento funcional calcular total llamadas	48
Tabla 12. Requerimiento funcional generar grafica	49
Tabla 13. Requerimiento funcional generar zoom	50
Tabla 14. Requerimiento no funcional aplicación web	51
Tabla 15. Requerimiento no funcional aplicación disponible	52
Tabla 16. Descripción de tablas modelo relacional	54

LISTA DE FIGURAS

	Pág.
Figura 1. Pasos para solucionar un problema	16
Figura 2. Funcionamiento Aplicación Web MVC	17
Figura 3. Áreas donde se ubica el problema	21
Figura 4. Fases Metodología Ágil	24
Figura 5. Gestion de requerimientos de desarrollo de aplicaciones	29
Figura 6. Caso de uso administrar	30
Figura 7. Caso de uso operar	30
Figura 8. Caso de uso ver resultados	31
Figura 9. Caso de uso visualizar llamadas	31
Figura 10. Diagrama de clases LineGrapher	32
Figura 11. Diagrama de componentes	34
Figura 12. Diagrama modelo base da datos	53

LISTA DE ANEXOS

	Pág.
Anexo A. DIAGRAMA DE PROCESOS / ACTIVIDADES	29
Anexo B. CASOS DE USO	30
Anexo C. DIAGRAMA DE CLASES	32
Anexo D. DIAGRAMA DE COMPONENTES	36
Anexo E. ESPECIFICACION DE REQUERIMIENTOS	35
Anexo F. DIAGRAMA DE BD/MODELO RELACIONAL	53
Anexo G. MANUAL DE USUARIO	55
Anexo H. MANUAL DEL SISTEMA	57

GLOSARIO

AJAX: Acrónimo de *Asynchronous JavaScript And XML*, utilización conjunta de diversas tecnologías, JavaScript y XML, para conseguir realizar peticiones asíncronas al servidor.

API: Del inglés *Application Programming Interface*, es un conjunto de comandos, funciones y protocolos que los programadores pueden utilizar en la construcción de software para un sistema operativo específico, se refiere a la interfaz externa que proporciona un determinado programa para ser invocado por otros.

Aplicación web: Aplicación informática cuyo medio de presentación es un navegador web.

Asíncrono: Significa que un proceso opera independientemente de otros procesos.

Bean: Objeto Java de almacenamiento de información volátil en aplicaciones web J2EE.

Blank: Se utiliza frecuentemente para referirse a aplicaciones web de ejemplo de incorporación y uso de una determinada librería, de forma que contienen el esqueleto mínimo.

Contenedor de servlets: Servidor web capaz de manejar el ciclo de vida de los servlets.

Cross-browser: Mantiene el aspecto y la funcionalidad de un navegador a otro.

Frames/Marcos: Divisiones en una página web, de manera que una página dividida en marcos referencia a su vez a un conjunto de páginas.

Framework/Marco de trabajo: Librería o herramienta para facilitar el desarrollo en una aplicación.

GPL: GNU *General Public License* es un tipo de licencia que acompaña al software libre y especifica qué el código fuente de dicho software puede ser copiado, modificado y distribuido libremente.

HTML: *HyperText Markup Language*, Lenguaje de Marcado de Hipertexto. Es el lenguaje más común en el que se escriben las páginas web, estos se pueden ver siempre dentro de las etiquetas "<", ">". HyperText Markup Language es el nombre que estas siglas representan, creado por Tim Berners-Lee en 1991 [Efraín and Marín, 1996].

J2EE: *Java Enterprise Edition*, es un entorno independiente de la plataforma de Sun para desarrollar, construir y desplegar aplicaciones web.

Java: Lenguaje de programación orientado a objetos desarrollado por Sun Microsystems.

JavaCC: Generador de *parsers* mas popular para el uso de aplicaciones escrito en Java.

JavaScript: Lenguaje de script utilizado para la creación de páginas web.

JSF: JavaServer Faces, framework estándar de Sun Microsystems para la capa de presentación de las aplicaciones web J2EE, simplifica el desarrollo mediante un enfoque centrado en el componente de desarrollo de interfaces de usuario.

JSP: JavaServer Pages, páginas web usadas en la capa de presentación de las aplicaciones web J2EE.

Servlet: Un servlet es una clase de Java que se utiliza para ampliar las capacidades de los servidores de aplicaciones de host que acceden por medio de un modelo de programación de petición-respuesta.

Lógica de negocio: Capa de un sistema que se encarga de realizar las tareas de negocio, es decir, aquellas para las que el sistema fue desarrollado en primera instancia.

MVC Modelo-Vista-Controlador: Patrón de diseño web.

Open-source: Código abierto, indica que el código fuente de una aplicación puede ser visto por cualquiera.

Parser: Aplicación que analiza un lenguaje cualquiera y realiza operaciones con los elementos que se van identificando y procesando.

Servidor web: Aplicación especial instalada en un ordenador que permite recibir y procesar peticiones de clientes remotos o locales de recursos locales.

Servlet: Clase en Java capaz de procesar una determinada petición web y generar el resultado correspondiente.

SQL: *Structured Query Language*, Lenguaje Estructura de Consultas. Es el lenguaje más utilizado para realizar consultas en clientes de Bases de Datos.

UML: *Unified Modeling Language*, Lenguaje de Modelado Unificado. Es un lenguaje de modelado ampliamente utilizado para describir varios aspecto de una aplicación, como puede ser funcional, de comportamiento, estructural, etc.

UTF-8: Sistema de codificación de caracteres.

WAR Web-Archive: Fichero comprimido que contiene una aplicación web J2EE.

XHTML Extended HTML: Lenguaje diseñado para remplazar a HTML. Proporciona las mismas funcionalidades y estructura pero es mucho más estricto al tener estructura XML.

XML EXtensible Markup Language: Lenguaje para transporte y almacenamiento de datos.

XSD: XML Schema Definition, Definición de Esquema XML. Describe la estructura de un fichero XML.

RESUMEN

Palabras Clave: Aplicación Web, Programación web, Bases de datos

Este trabajo de proyecto de grado se realizó con el objetivo de convertir en realidad una idea que surgió como la solución a una necesidad donde se plantea la implementación de una aplicación web la cual pretende mostrar de una forma más amigable una gráfica en el tiempo para analizar y monitorear el número de llamadas simultaneas entrantes en una empresa de Contact Center.

La principal característica de esta aplicación web es la facilidad que se tiene para visualizar la información gracias a un conjunto de componentes gráficos que permiten ayudar en la lógica de negocio, el acceso a los datos y la presentación de la información.

El presente proyecto de ingeniería de sistemas se desarrolló en 3 campos. El primero la Programación Web siendo este la base principal del proyecto, se apoyó posteriormente en el campo de Diseño Web y finalmente toda su información en el campo de Base de Datos.

Para la realización de esta aplicación fue necesario apoyarse en una metodología especializada en aplicaciones Web. OOHDM (Object Oriented Hypermedia Design Methodology), esta a su vez diseñada en el lenguaje de modelamiento UML que permitio el desarrollo de todos los diagramas que servirian como base para la programación. Para la implementación de la aplicación se utilizó como lenguaje Java EE con el Framework de Primefaces, y se implementó en el motor de base de datos SQL Server 2008.

ABSTRACT

Key Works: Web Application, Web Programming, Databases

This degree project work was performed with the aim to realize an idea that emerged as the solution to a need which proposes the implementation of a Web application which pretend to show a friendlier graph in the time to analyze and control the number of simultaneous incoming calls on a contact center company.

The main feature of this web application is the ease you have to display information through a set of graphic components that help enable business logic, data access and presentation of information.

This project was developing systems engineering in 3 fields. The first Web Programming being the main base of the project, further support in the field of Web Design and finally all their information in the database field.

For the realization of this application was necessary to rely on a method specialized Web applications. OOHDM (Object Oriented Hypermedia Design Methodology), this in turn designed in the UML modeling language that allowed the development of all diagrams that would serve as the basis for programming. For the implementation of the application was used Java EE Framework Primefaces, and implemented in SQL Server 2008.

1 INTRODUCCIÓN

El tráfico en la red de telefonía permite día a día tomar decisiones acerca de cómo se comporta la entrada y salida de llamadas en un momento determinado esto con el fin de analizar los diferentes eventos y comportamientos que se presentan.

La tecnología ha ejercido una gran influencia en las áreas de conocimiento, es por esto que los Contacts Centers se han convertido en una herramienta de vital importancia para pequeñas y grandes empresas con el fin de ofrecer los elementos necesarios para tener una buena comunicación y relación con los clientes.

Las empresas de Contact Center viven en un mundo global competitivo donde cada vez se necesitan aplicaciones para satisfacer las necesidades de negocio. Con el avance de las tecnologías web y la Internet, se han abierto nuevas oportunidades para el desarrollo de aplicaciones empresariales; permitiendo el uso de las nuevas tecnologías en el desarrollo de aplicaciones mucho más robustas, escalables y con un mayor rendimiento.

Las aplicaciones web facilitan el análisis de datos de una forma más amigable donde cada vez se hace más necesario la presentación y utilización de los resultados.

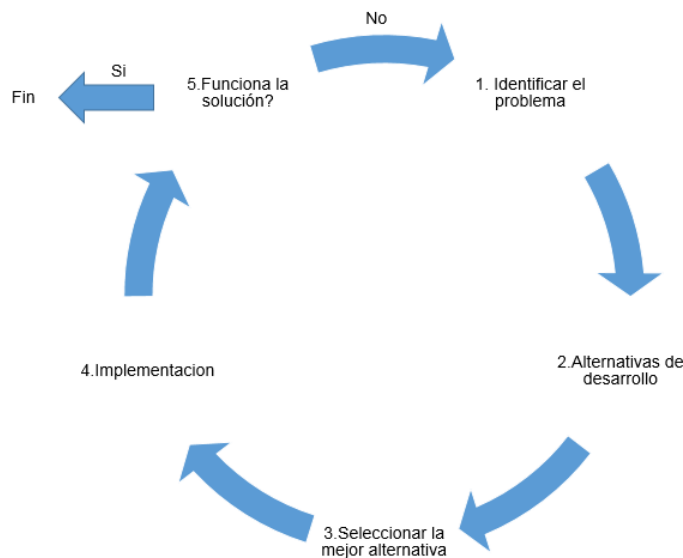
2. PLANTEAMIENTO DEL PROBLEMA

Como saber el número de llamadas simultaneas en un día?Cuál es el aporte de la tecnología en los Contact Centers? Como facilitar el análisis y la presentación de los resultados?

El presente proyecto busca implementar una aplicación web para el uso de la plataforma de Contact Center existente en la empresa, dentro de sus actividades administra el ingreso y la salida de llamadas, esto con el fin de ser el mejor aliado estratégico en Contact Center y tercerización de procesos.

Se tiene como motivación la medición del número de llamadas simultaneas entrantes de diferentes compañías y grupos de trabajo, donde se especifican los criterios de medición de flujo de tráfico de una manera sencilla independientemente del resto del tráfico. Se definen los siguientes pasos para el planteamiento del problema (Figura 1).

Figura 1. Pasos para el planteamiento del problema



Fuente. El autor

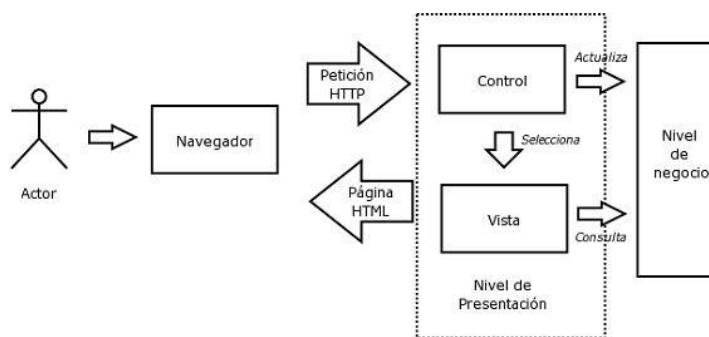
Actualmente la empresa realiza informes en hojas de cálculo y no cuenta con una aplicación web que facilite el análisis y la presentación de los mismos.

Para entender el planteamiento del problema es necesario aclarar en primer lugar el contexto del mismo. En el desarrollo e implementación de aplicaciones web se utilizan frecuentemente frameworks o marcos de trabajo para ayudar en la lógica de negocio, el acceso a datos y la presentación de la información.

Uno de los frameworks más utilizados hoy en día para la vista es JSF (JavaServer Faces)¹, el cual está basado en el uso de componentes para presentar datos y funcionalidad encapsulada. En la actualidad, existen numerosas librerías de componentes JSF, cada una con su propio conjunto de componentes, su forma de configuración y modo de uso.

Estos marcos que facilitan la presentación de los datos son conocidos como “marcos para la vista”, haciendo referencia a la parte concreta que resuelven del patrón de diseño web MVC (Modelo-Vista-Controlador)².

Figura 2. Funcionamiento Aplicación Web MVC



Fuente. <http://www.cii murcia.es/informas/ene05/articulos/imagenes/ArquitecturaDeTresCapas.jpg>

2.1 RESULTADOS

- Desarrollo e implementación de una aplicación a través de la utilización de software libre: Herramienta de desarrollo Java EE³, Oracle como motor de base de datos y PrimeFaces como Framework o Marco de trabajo.
- Se ha facilitado la toma de decisiones en cuanto a aumentar o disminuir en número de canales en los servidores de entrada (Inbound).
- Manuales del sistema y de usuario sobre la implementación de la aplicación web
- Se ha informado al personal del área de Plataformas de Contact Center sobre el manejo de la aplicación.

¹ JavaServer Faces, <http://docs.oracle.com/javaee/6/tutorial/doc/bnaph.html>

² Modelo-Vista-Controlador, <http://www.desarrolloweb.com/wiki/mvc-modelo-vista-controlador.html>

³ <http://www.oracle.com/technetwork/java/javaee/overview/index.html>

2.2 JUSTIFICACION

Este proyecto, no sólo beneficiará al área de tecnología, sino que le proporcionará una herramienta a la plataforma de Contact Center para que tenga una mejor forma de visualizar la información.

Por otra parte, se garantiza la aplicación de la ingeniería de sistemas en el área de tecnología, a través de la interacción de personas, datos, procesos, información como una solución en el entorno de trabajo.

Finalmente, la tecnología en Colombia debe recurrir a las TIC, Tecnologías de Información y Comunicación, para seguir los pasos del desarrollo y fortalecer las metodologías y estrategias del sector tecnológico.

3. OBJETIVOS DEL PROYECTO

3.1. OBJETIVO GENERAL

Desarrollar e implementar una aplicación web que permita el análisis y monitoreo de llamadas entrantes en una empresa de Contact Center

3.2. OBJETIVOS ESPECIFICOS

- Calcular el número de llamadas simultáneas de entrada (Inbound).
- Identificar patrones de comportamiento en las llamadas de entrada.
- Desarrollar la aplicación web teniendo en cuenta el número de llamadas simultáneas entrantes.
- Realizar la documentación de la aplicación.

4. MARCO REFERENCIAL

4.1. MARCO CONCEPTUAL

El proceso de ingeniería de sistemas es denominado ingeniería de procesos de negocio cuando el contexto del trabajo de ingeniería se enfoca a una empresa. Cuando hay que construir un producto, el proceso se denomina ingeniería de producto⁴.

Tanto la ingeniería de proceso de negocio como la de producto intentan poner orden al desarrollo de sistemas basados en computadoras. Aunque cada una se aplica en un dominio de aplicación diferente, ambas intentan poner al software en su contexto.

La ingeniería de Sistemas permite que se puedan definir los objetivos generales del Sistema; de igual manera se puedan identificar los papeles del hardware, software, personas, bases de datos, procedimientos, y otros elementos del sistema; en este mismo sentido también se puede observar los requerimientos funcionales los cuales son identificados, analizados, especificados, modelados, validados y gestionados.

Un Contact Center (también conocido como centro de interacción con el cliente o centro de contacto) es un punto central en una empresa donde todos los contactos son administrados. Por lo general incluye uno o más centros de atención telefónica en línea, pero puede incluir otros tipos de contacto con el cliente como correo electrónico, sitio web y chats.

Es generalmente parte de la gestión global de los clientes típicamente una moderna solución de comunicaciones unificadas con un software especializado, es considerado como un elemento importante en la empresa.

Una aplicación web es cualquier aplicación que es accedida vía web por una red como internet o una intranet.

Una de las ventajas de las aplicaciones web cargadas desde internet (u otra red) es la facilidad de mantener y actualizar dichas aplicaciones sin la necesidad de distribuir e instalar un software, también la posibilidad de ser ejecutadas en múltiples plataformas.

⁴ PRESSMAN. Ingeniería del Software: un enfoque práctico. McGraw Hill. Madrid, 2002. 5ª Edición. Pág. 10

Se presentan las principales características⁵:

- El usuario puede acceder fácilmente a estas aplicaciones empleando un navegador web (cliente) o similar.
- Si es por internet, el usuario puede entrar desde cualquier lugar del mundo donde tenga acceso.
- Pueden existir miles de usuarios pero una única aplicación instalada en un servidor, por lo tanto se puede actualizar y mantener una única aplicación y todos los usuarios verán los resultados inmediatamente.

En cuanto a el servidor de aplicaciones es un software que proporciona aplicaciones a los equipos o dispositivos del cliente, por lo general a través de Internet y utilizando el protocolo http. Los servidores de aplicaciones se distinguen de los servidores web por el uso extensivo del contenido dinámico y por su frecuente integración con bases de datos.

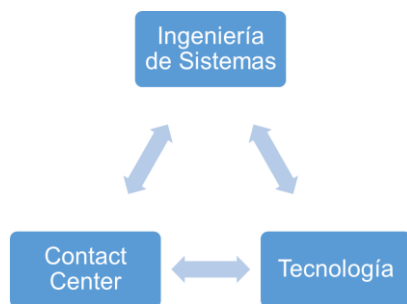
El concepto de servidor de aplicaciones está relacionado con el concepto de sistema distribuido. Un sistema distribuido, en oposición a un sistema monolítico, permite mejorar tres aspectos fundamentales en una aplicación: la alta disponibilidad, la escalabilidad y el mantenimiento.

4.2. MARCO TEORICO

El proyecto planteado anteriormente, se encuentra dentro de las áreas del conocimiento que definen los fundamentos teóricos para dar solución a una necesidad, estas áreas se encuentran dentro del campo de la Ingeniería de Sistemas.

A continuación, se definen los principales conceptos relacionados con el proyecto:

Figura 3. Áreas donde se ubica el problema



Fuente. El autor

⁵ <http://www.alegsa.com.ar/Dic/aplicacion%20web.php>

4.2.1 BASES DE DATOS

Una base de datos es un conjunto de datos tratados como una unidad. El propósito de una base de datos es almacenar y recuperar información relacionada, son una alternativa al uso de múltiples archivos convencionales. Los archivos, son colecciones de datos semejantes. Éstas, son posibles gracias al uso de software especializado en gestión de bases de datos⁶.

4.2.3 TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIONES (TIC)

Comprenden todos los usos de la tecnología digital existentes para ayudar a individuos, empresas y organizaciones al buen uso de la información.

Incluyen cualquier producto que vaya a almacenar, recuperar, manipular, transmitir o recibir información electrónicamente en un formato digital. Por ejemplo, los ordenadores personales, televisión digital, correo electrónico.

4.2.4 SOFTWARE LIBRE

Software libre significa que el software respeta la libertad de los usuarios y la comunidad. En términos generales, los usuarios tienen la libertad de copiar, distribuir, estudiar, modificar y mejorar el software. Con estas libertades, los usuarios (tanto individualmente como en forma colectiva) controlan el programa y lo que hace.

Cumple con cuatro libertades: la libertad de usar el programa, con cualquier propósito (libertad 0); estudiar el funcionamiento del programa, y adaptarlo a sus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto; la libertad de distribuir copias (libertad 2); finalmente, la libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie (libertad 3).

La libertad para usar un programa significa las libertades para cualquier persona u organización de usarlo en cualquier tipo de sistema informático, para cualquier clase de trabajo, y sin tener obligación de comunicárselo al desarrollador o a alguna otra entidad específica⁷.

⁶ WHITTEN, Jeffrey, L. Análisis y Diseño de Sistemas de Información. Nomos S.A. Colombia: 1999. p.606

⁷ GNU Operating System – Free Software Foundation

El software libre como el que se usó en la implementación (Tabla 1), dispone de la Licencia Pública General GPL, que permite hacer uso de todas las libertades mencionadas anteriormente.

Tabla 1. Software utilizado en la implementación

Software	Descripción
1. Java	Lenguaje de programación para construir la aplicación Web.
2. PrimeFaces	Framework o Marco de trabajo utilizado en la aplicación.
3. Oracle	Motor de base de datos compatible con Java.
4. Apache	Servidor web, gestiona el funcionamiento de la aplicación.

Fuente. El autor

4.2.5 REDES

Las redes informáticas y de comunicaciones son disposiciones técnicas que interconectan ordenadores y periféricos de manera que se puedan intercambiar datos y compartir recursos técnicos. En ocasiones recibe el nombre de arquitectura distribuida de sistemas.

4.2.6 VIABILIDAD

De un proyecto, se conoce como el análisis o el estudio que intenta predecir el eventual éxito o fracaso de un proyecto, se refiere al estudio de factibilidad comercial, técnica, legal, de gestión, de impacto ambiental y financiera. Cada uno de estos elementos determina que un proyecto se concrete o no en la realidad.⁸

⁸ RAMÍREZ, Cardona Carlos. Fundamentos en Administración. Ecoe. Bogotá: 1999. p.42.

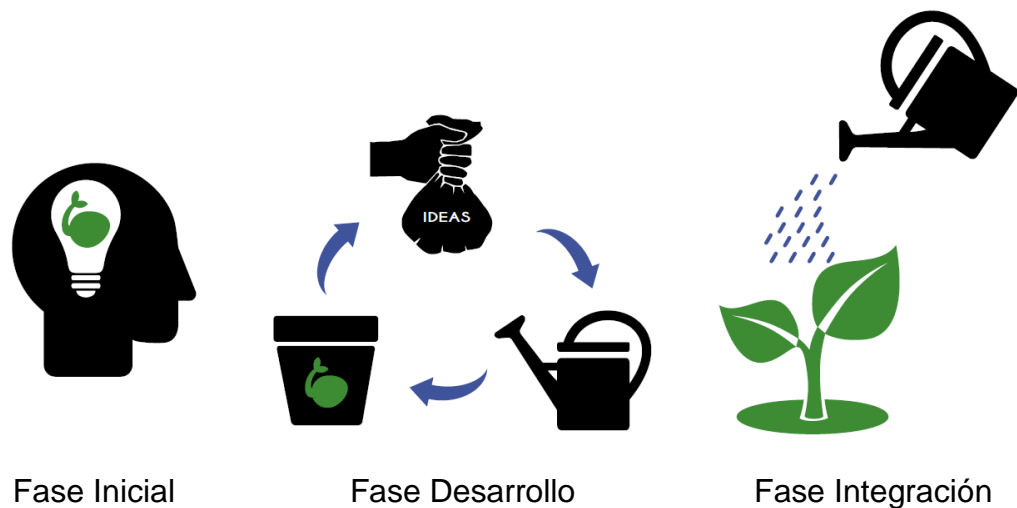
5. METODOLOGIA PROPUESTA

El desarrollo del proyecto se realizó bajo la metodología ágil la cual permite trabajar con las siguientes características:

1. Es una alternativa a la gestión tradicional de proyectos, se utiliza normalmente en el desarrollo de software. Está preparada para adecuarse rápidamente a cambios y mejoras, diseñada para trabajar muy de cerca, realizar entregas para utilizarlas en periodos cortos de tiempo y poder así recibir una retroalimentación constante.
2. Ayuda a los equipos a que respondan a los imprevistos a través de incrementos iterativos de trabajo, más conocidos como sprints.
3. Las metodologías ágiles son una alternativa al desarrollo en cascada, o el desarrollo secuencial tradicional.

Una metodología es una forma, manera o método de llevar a cabo un proceso. En este caso la elaboración del proyecto “Aplicación Web para el análisis y monitoreo de llamadas entrantes en una empresa de Contact center”.

Figura 4. Fases Metodología Ágil



Fuente. El autor

A continuación se explica cada una de las fases que se utilizaron en el desarrollo del proyecto, se muestran en la Figura 4.

Fase Inicial

En esta fase se entienden y se adecuan las necesidades, se hace levantamiento de información y se plantean las principales necesidades a suplir, se hace una conceptualización de la solución general donde se definen los pasos cortos o pequeños con funciones específicas, finalmente se hace la retroalimentación.

Se hizo un análisis del contexto del proyecto apoyado del desarrollo web investigando los marcos de trabajo relacionados con la parte de la vista del patrón de diseño MVC, centrándose en JavaServer Faces y un amplio conjunto de librerías de componentes para este marco de trabajo.

Fase Desarrollo

Se trabaja por iteraciones cortas donde se busca siempre mejorar el trabajo constantemente y redefinir funcionalidades, conforme a esto se avanza hacia una retroalimentación constante, la solución a desarrollar se determinó en base al problema localizado.

Se llegó a la conclusión de implementar una aplicación web era la mejor opción, de tal forma que se pudiera mostrar las funcionalidades de JSF.

Fase Integración

Es la fase del lanzamiento de la aplicación, también tiene que hacerse una retroalimentación constante de las actividades y procesos para realizar la integración de mejoras y nuevas soluciones.

En esta fase se valida que la solución propuesta resuelva correctamente el problema planteado. Para esto se utilizan una serie de métodos y se analizan los resultados obtenidos.

6. DESARROLLO DEL PROYECTO

6.1. RECURSOS

La siguiente lista contiene el hardware y software que se utilizó en el desarrollo de la aplicación. Se estimó que era suficiente para llevar a cabo el desarrollo del proyecto.

Hardware:

- Computador Lenovo Intel Core I5 para el desarrollo de la aplicación.
- Impresora
- USB Kingstone 4GB

Software:

- Sistema Operativo Windows 7
- LibreOffice (Procesador de texto y hojas de cálculo).
- Herramienta de modelado UML Astah Community
- Herramienta de modelado de base de datos Creately⁹
- Modelador de procesos Bizagi¹⁰
- IDE de desarrollo Netbeans 7.3
- Base de Datos Oracle 11g
- SQL Developer
- Servidor de Aplicaciones Tomcat 7.0.40

⁹ <http://creately.com/>

¹⁰ <http://www.bizagi.com/>

6.2 IMPLEMENTACIÓN Y PRUEBAS

6.2.1 JSF

En un nivel abstracción, JSF es una implementación del Modelo Vista Controlador (MVC)¹¹. Esta arquitectura se refiere a la interfaz de usuario, la lógica de negocio, y su conector que permite la comunicación entre la interfaz de usuario y la lógica de negocio. JSF es un componente para las aplicaciones web basadas en Java. El marco proporciona las siguientes características:

- Especificar la navegación de la página
- Componentes de interface de usuario estandar como input fields , botones y los links
- Validación de entrada de usuario
- Manejo fácil de errores
- Gestión de Java beans
- Gestión de eventos
- Soporte multiidioma

6.2.2 Primefaces

Es un componente para JavaServer Faces (JSF) de código abierto que cuenta con un conjunto de componentes que facilitan y ayudan a la creación de las aplicaciones web. Primefaces es una librería open source que está bajo la licencia de Apache License V2. Una de las ventajas de utilizar Primefaces, es que permite la integración con otros componentes como por ejemplo RichFaces¹².

PrimeFaces se divide principalmente en tres módulos: El primero es el set de componentes para la interfaces de usuario, el segundo módulo llamado Optimus utiliza Guice para poder crear managed beans utilizando anotaciones, simplificar la navegación entre páginas e integrar PrimeFaces con JPA, transacciones y más. Y un tercer módulo llamado FacesTrace permite monitorear aplicaciones JSF.

Versiones

- Primefaces 1: Trabaja con JSF 1.2
- Primefaces 2 y 3: Trabaja con JSF 2

¹¹ http://es.wikipedia.org/wiki/Modelo_Vista_Controlador

¹² RichFaces. - Es una biblioteca de código abierto basada en Java para crear aplicaciones web con Ajax.

Se enumeran los pasos para el desarrollo de la aplicación:

1. Crear las páginas JSP
2. Definir una regla de navegación
3. Crear un managed bean
4. Crear un fichero de properties
5. Editar páginas JSP
6. Crear un fichero index.jsp
7. Compilar la aplicación
8. Desplegar y ejecutar la aplicación

6.3 PRUEBAS

Pruebas unitarias: Consisten en probar el correcto funcionamiento de un módulo de código dado, se hizo la depuración de las líneas de código.

Pruebas de integración: Se centran en comprobar que los diferentes módulos que componen un subsistema se integran correctamente, con el correcto funcionamiento de la aplicación.

Ejecución de pruebas: Se llevó a cabo la ejecución de las pruebas diseñadas y planificadas por el área de soluciones informáticas.

6.3.1. Especificación de Caso de Prueba:

Seguridad:

Descripción: Al iniciar la aplicación aparecerá un formulario de acceso al sistema, el usuario que desee ingresar a la aplicación deberá tener un usuario y una contraseña.

Además, el sistema al ser una aplicación web puede facilitarse para que pueda acceder a las páginas directamente por ingreso de la url por lo que hemos desarrollado un módulo de seguridad, el mismo que empaqueta la aplicación para que no se pueda acceder de manera aleatoria.

Disponibilidad:

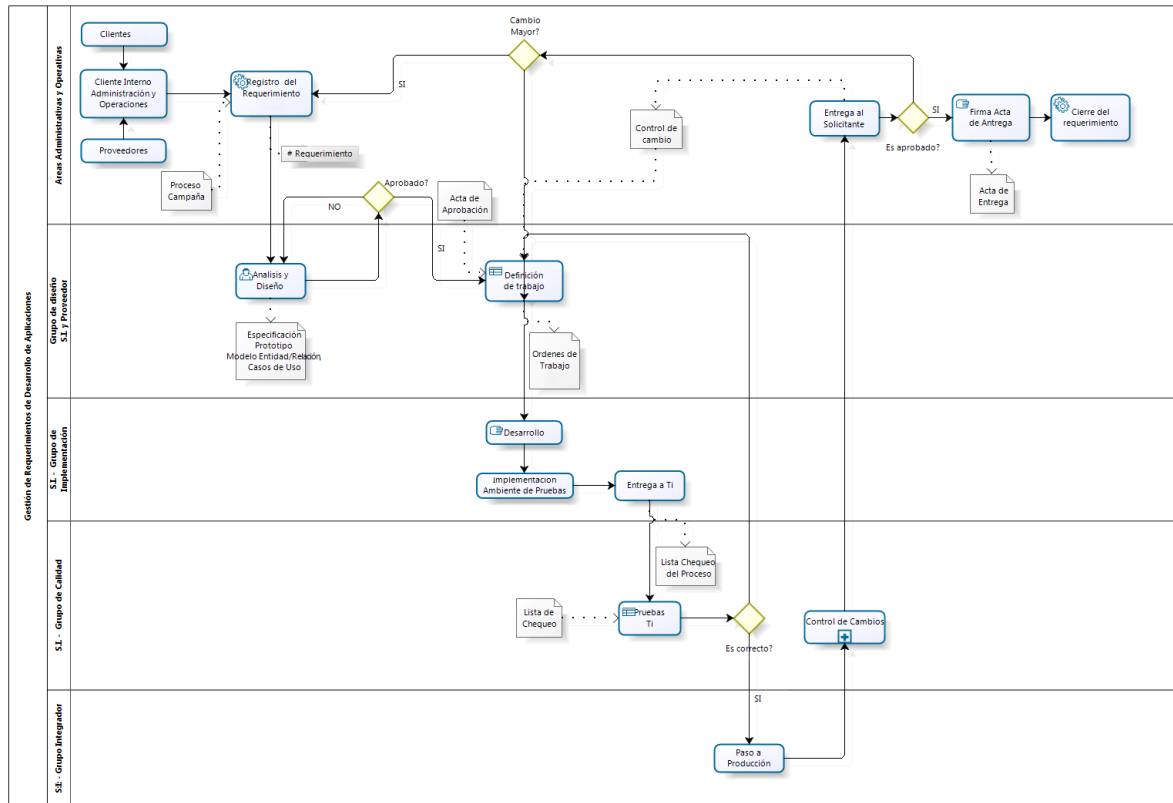
Descripción: Se accedió a la aplicación desde varios ordenadores para evitar caídas en el servidor esto con el fin de prevenir posibles fallos a futuro.

Desempeño:

Descripción: Se estimaron tiempos en las consultas evaluando diferentes escenarios esto con el fin de presentar la información en el menor tiempo posible.

ANEXO A: DIAGRAMA DE PROCESOS / ACTIVIDADES

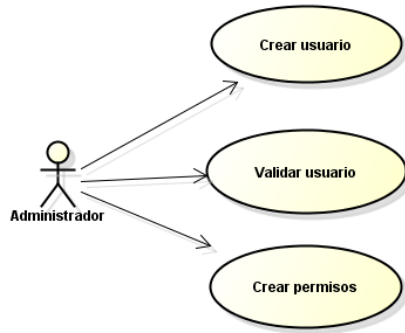
Figura 5. Gestion de requerimientos de desarrollo de aplicaciones



Fuente. El autor

ANEXO B: CASOS DE USO

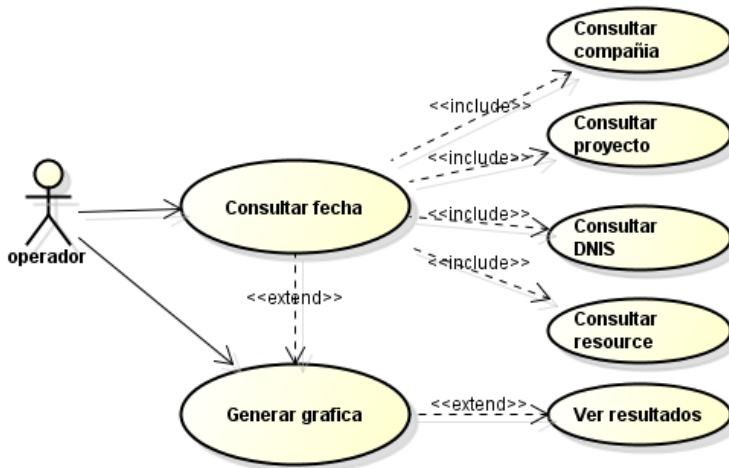
Figura 6. Caso de uso administrar



Fuente. El autor

Descripción: El actor principal Administrador tiene la responsabilidad de crear los usuarios y perfiles que van a tener acceso a la aplicación, también se encarga de validar los usuarios creados y los permisos que tiene cada uno sobre las diferentes funcionalidades.

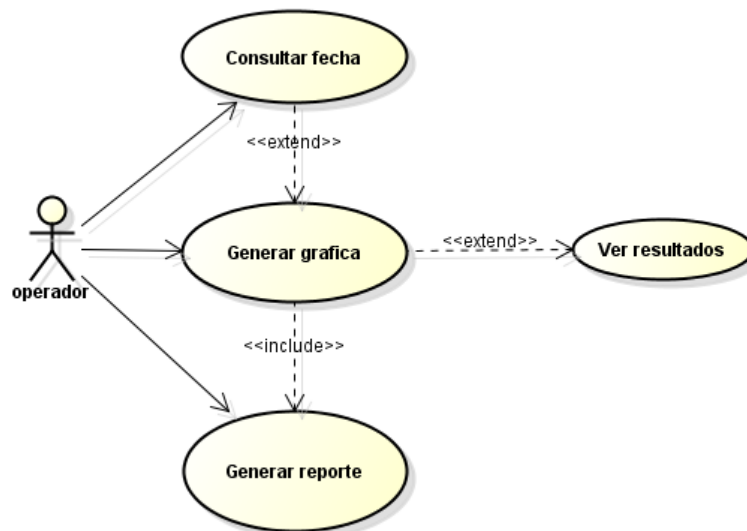
Figura 7. Caso de uso operar



Fuente. El autor

Descripción: El operador es el que va a ejecutar las funcionalidades que tiene la aplicación en este caso es el que va a dar los valores de entrada como fecha, compañía, proyecto, DNIS para visualizar la gráfica.

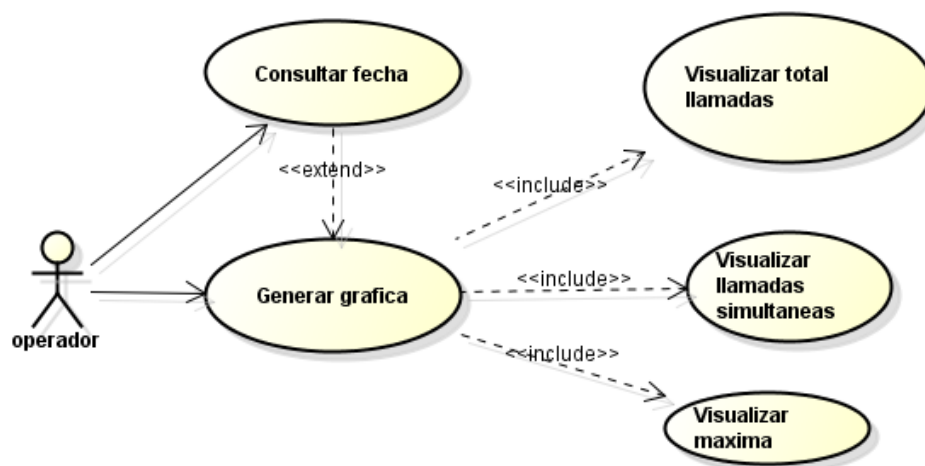
Figura 8. Caso de uso ver resultados



Fuente. El autor

Descripcion: El operador tiene la posibilidad si es necesario de generar un reportte ya sea en formato de Excel o en pdf.

Figura 9. Caso de uso visualizar llamadas

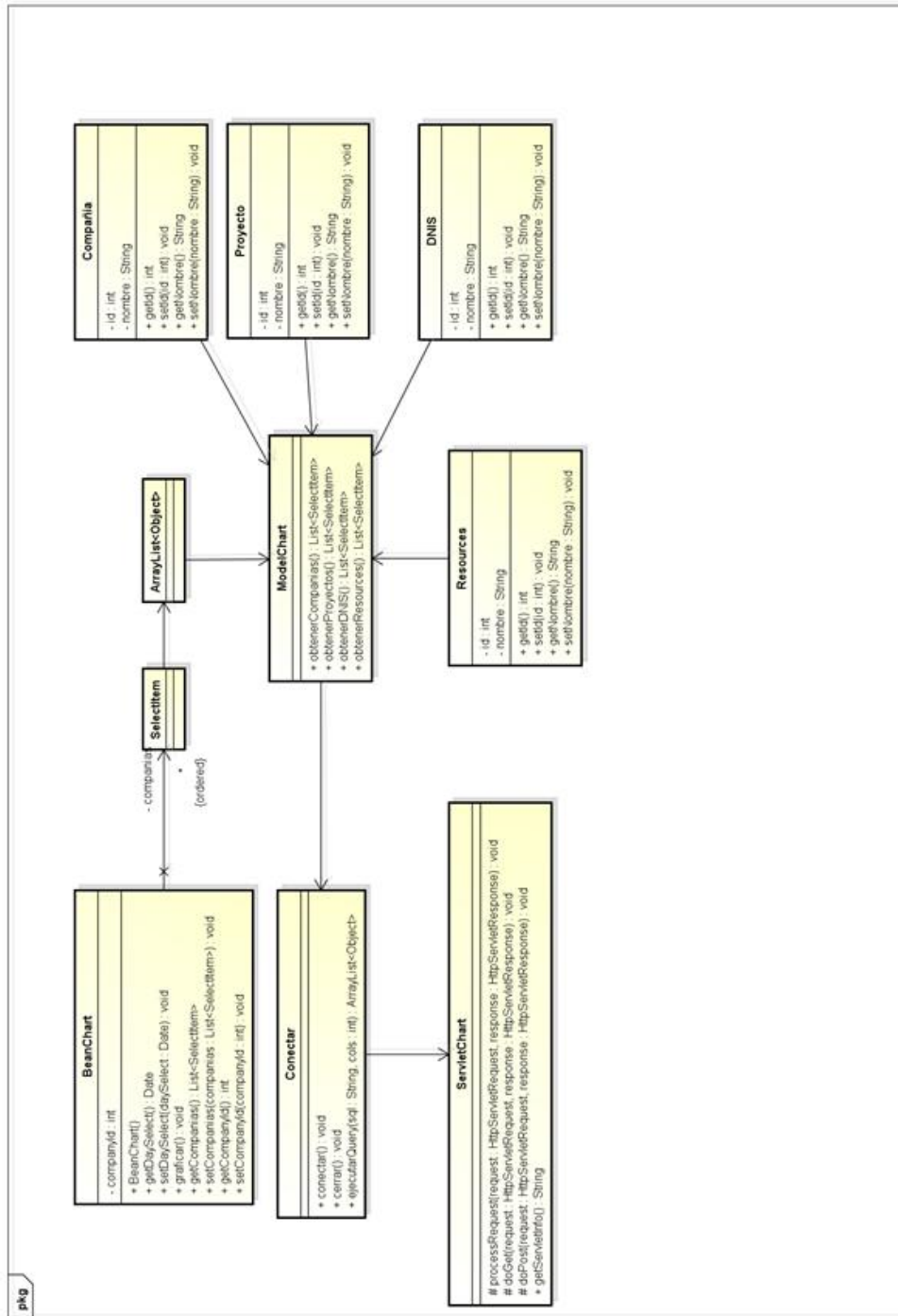


Fuente. El autor

Descripcion: El operador puede evaluar patrones de comportamiento teniendo en cuenta el número de llamadas.

ANEXO C: DIAGRAMA DE CLASES

Figura 10. Diagrama de clases LineGrapher



Fuente. El autor

La aplicación se compone de 5 clases principales, cada una de ellas tiene un papel importante en la implementación y en el funcionamiento de la misma, a continuación se explican cada una de ellas:

Conectar: Se encarga de realizar la conexión a la base de datos y de realizar las consultas para obtener la información requerida por el usuario.

ModelChart: Obtiene los nombres de las compañías, proyectos, DNIS y resources de la base de datos para poder desplegarlas en el menú principal de la aplicación.

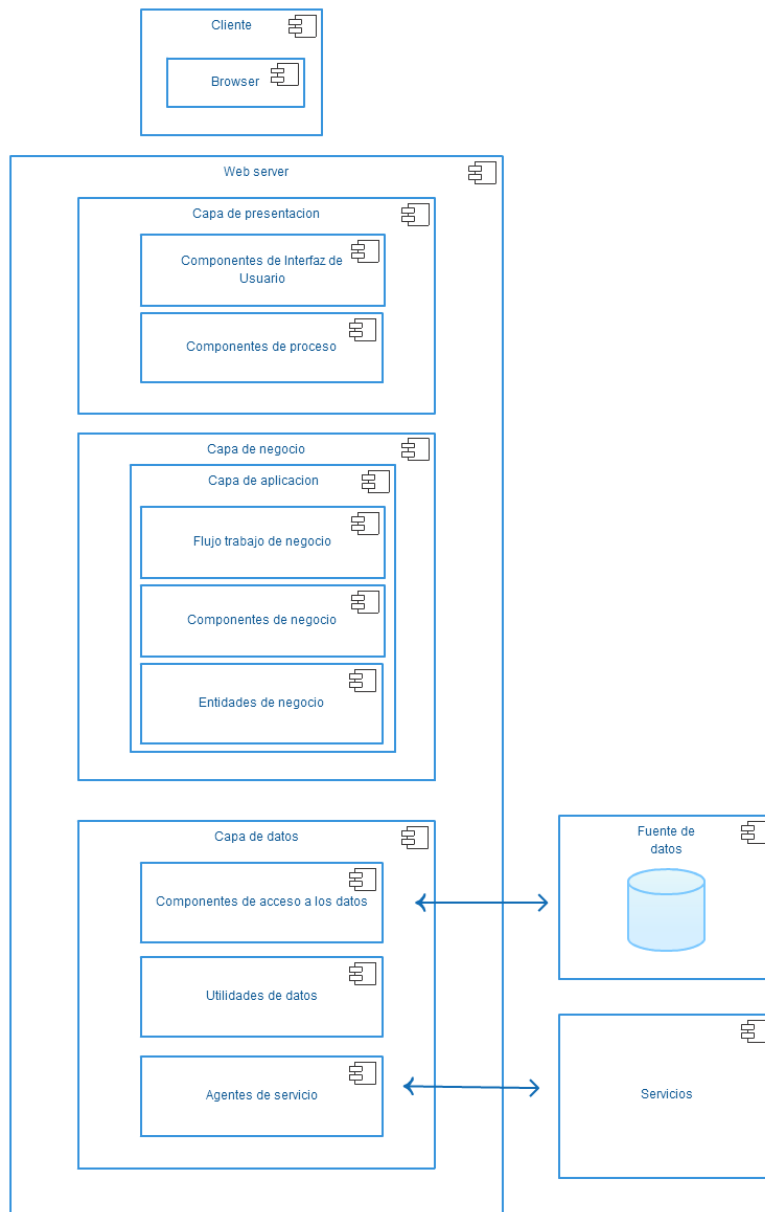
Entidades: Controla el encapsulamiento de los datos y del tipo de información que se esta manejando en la aplicación por medio de los “getters and setters”.

ServletChart: Recibe peticiones de uno o más clientes, a través de esta clase los métodos de HttpServlet envían información a los clientes que les han pedido algún servicio.

BeanChart: Esta clase define un conjunto de convenciones (design patterns) que el IDE usa para inferir qué métodos corresponden a las propiedades de la clase.

ANEXO D: DIAGRAMA DE COMPONENTES

Figura 11. Diagrama de componentes



Fuente. El autor

En el diagrama de componenes presentado se evaluan las diferentes arquitecturas de la aplicación agrupando cada una de ellas en 3 capas diferentes cada una con un rol especifico en el funcionamiento de la misma.

ANEXO E: ESPECIFICACION DE REQUERIMIENTOS

Propósito

Presentar oficialmente la documentación detallada de la descripción de la aplicación web, los requerimientos de interfaz externa, requerimientos funcionales y no funcionales para su implementación.

Alcance

Describir el funcionamiento de la aplicación web con los documentos entregables establecidos y corregidos.

Tabla 2. Glosario de términos

TÉRMINO	DEFINICIÓN
Arquitectura cliente servidor	Es un modelo para ejecutar tareas de computación, en conjunto con mínimo un servidor que presta unos servicios y uno o más cliente conectados que hace peticiones sobre ellos
Stakeholder	Interesados en el proyecto
Interface (de usuario)	Es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.
Aplicación web	Una aplicación web es cualquier aplicación que es accedida vía web por una red como internet o una intranet (red de ordenadores privados que utiliza tecnología Internet para compartir dentro de una organización).
Programador	Es la persona responsable de tomar decisiones en cuanto al desarrollo de la aplicación, sin distinción entre analistas, diseñadores o programadores. Los programadores diseñan, programan y realizan las pruebas.
Tutor	Organiza y guía las reuniones además asegura condiciones adecuadas para el proyecto
Parametrizar	Establecer los valores a utilizar en los formatos y comportamientos del sistema.
Plataformas	Es un sistema que sirve como base para hacer funcionar determinados módulos de hardware o de software con los que es compatible.
Browser	Un navegador o navegador web (del inglés, web browser) es una aplicación que opera a través de Internet, interpretando la información de archivos y sitios web para que estos puedan ser leídos.
Sistema	Hace referencia a los conjuntos estructurados y organizados que se encuentran en el proyecto de implementación.

Fuente. El autor

Descripción del sistema

La empresa de Contact Center proporciona un entorno de desarrollo que contiene un conjunto de módulos para la implementación de una aplicación web con arquitectura cliente servidor, donde se presta la atención y seguimiento a los servidores de entrada (Inbound) y de salida (Outbound), esto con el fin de facilitar el análisis de la cantidad de llamadas entrantes en las diferentes campañas y grupos de trabajo.

Implementándose exclusivamente para la empresa de Contact Center, en el área de tecnología la información relacionada con el proyecto, será consultada y procesada, no se tendrá en cuenta ningún tipo de información que no se encuentre directamente relacionada con la aplicación, tampoco se considerará la información proveniente de las llamadas salientes, es posible el uso de cualquier herramienta funcional enfocada a tareas puntuales que cumplan con los requerimientos mínimos exigidos.

Funciones del sistema o producto

Satisfacer la presentación, procesamiento y actualización de la información requerida para su gestión y seguimiento a la empresa de Contact Center.

Tabla 3. Usuarios del sistema

Usuario	Descripción	Experiencia con aplicaciones informáticas	Expectativas de usabilidad
Operador	Maneja datos sensibles y tiene acceso a la mayoría de la información del Contact Center. Puede consultar, y generar graficas siempre y cuando tenga el correspondiente acceso autorizado.	Nivel de experiencia alto, desempeña la mayor parte del tiempo independientemente, solo solicita la ayuda cuando ocurren situaciones fuera de su autoridad.	El sistema es fácil de manejar, en donde las peticiones para generar gráficas se encuentran instintivamente.
Administrador de servicios	El súper-usuario encargado de asignar autorizaciones y restricciones en el sistema a los demás usuarios.	Nivel experto, se desenvuelve con naturalidad y conoce todas las características y funcionalidades del sistema.	Fácil de usar, visualmente agradable y rápido acceso a las aplicaciones que él requiera.

Fuente. El autor

Restricciones del sistema

1. Debe desarrollarse el proyecto en su totalidad para ser implementado en el periodo que dure la práctica empresarial
2. La aplicación debe funcionar en web
3. Restringir las consultas con un rango de tiempo no superior a un día
4. Cumplir con las características mínimas de conectividad

Suposiciones

1. El costo y presupuesto del proyecto es aprobado y no existe ningún problema de sobrecostos.
2. La aplicación web es flexible y fácil de mantener.
3. La disponibilidad y velocidad de navegación es óptima cumpliendo con los parámetros de disponibilidad y conectividad.
4. La interacción de clientes, operadores y administradores con la aplicación web es satisfactoria

6.2.6.1 REQUERIMIENTOS DE INTERFACE EXTERNA

Interfaces de usuario

El objetivo técnico de las interfaces de usuario es el de la adaptación entre la complejidad de los sistemas y las capacidades del ser humano.

Cumpliendo con:

- Naturalidad: La aplicación web debe cumplir con el estándar de desarrollo de las demás aplicaciones de la empresa
- Facilidad de aprendizaje y uso: dos aspectos que no siempre van unidos
- Consistencia: La interfaz debe mantener uniformidad en cuanto a estilo, vocabulario, etc.

Características:

- Naturalidad

Una interfaz es natural, cuando es agradable a la vista. Todo usuario tiene:

- Una forma de actuar
- Una forma de organizarse
- Un vocabulario propio para las tareas habituales
- Un entorno que ya domina, al que está acostumbrado y del que, tal vez, le sea difícil de salir.

- Facilidad de aprendizaje y uso

Proporcionar al usuario una aplicación que facilite la presentación de datos de una forma más amigable y entendible para poder así analizar el comportamiento de las llamadas entrantes. Para disfrutar de esta característica, la interfaz debe incorporar:

- Administración de perfiles de usuario: Según el grado de perfil, la interfaz ejecutará unas acciones u otras.
- Mecanismos de realimentación que proporcione al usuario información sobre la ejecución actual del trabajo.
- Sistemas de ayuda: Tratan de evitar que el usuario tenga que acceder a los manuales para resolver una duda puntual. Los mejores sistemas de ayuda son los que se denominan “sensibles al contexto”.

- Consistencia

Debe mantenerse una uniformidad a lo largo de toda la extensión de la interfaz: modo de operación, diseño, etc. Si cada componente actúa con distinta filosofía, obliga al usuario a cambiar la mentalidad de trabajo.

Interfaces adaptativas:

Las interfaces adaptativas son las que se adaptan a diferencias o cambios que existen o pueden tener lugar en la población de usuarios de un sistema informático.

Al decir “diferencias” se refiere a aquellas que existen entre distintos usuarios en un momento dado (ej. usuarios de distintos países en este caso regiones), mientras que los “cambios” se producen para un mismo usuario a lo largo del tiempo (ej. usuario que pasa de ser “novato” a “experto”).

Razones para el comportamiento adaptativo:

- El sistema es usado por usuarios con diferentes requerimientos: usuarios de diferentes países o idiomas, usuarios con minusvalías, etc.
- El sistema es usado por usuarios con requerimientos cambiantes: usuarios que evolucionan de “normal” a “experto”, etc.
- El usuario trabaja en varios entornos: consistencia entre diferentes aplicaciones, Word, Excel, etc.
- El usuario trabaja en un entorno cambiante: cambios impuestos por organismos oficiales, etc.

Diseño de interfaces adaptativas:

Para realizar el diseño de una interfaz adaptativa es preciso responder a tres preguntas:

¿Qué?: En primer lugar hay que determinar qué aspectos de la interfaz van a ser adaptables a las diferencias o cambios considerados.

- La presentación al usuario de las entradas al sistema, seleccionando entre varios estilos de interacción (menú, ordenes, etc.).
- Corregir errores, entradas inapropiadas y sistemas de ayuda activa, para presentarle al usuario la información más apropiada.
- Presentación de información del sistema al usuario. Seleccionar el estilo de presentación y la información que queremos ver.
- La adaptación de funcionalidades: que el sistema se encargue de las tareas rutinarias y el usuario de las tareas creativas (empleo de macros)

¿Cuándo?: También es importante saber en qué momento han de realizarse tales adaptaciones.

- Antes de la primera sesión: en el momento de la instalación (ej. idioma)
- Durante las sesiones: puede ser a petición del usuario o de forma continua y automática.
- Entre las sesiones: se tiene en cuenta información recogida en sesiones anteriores.

¿Cómo?: Finalmente es preciso tener en cuenta los diferentes métodos y técnicas que podemos utilizar para hacer efectivas tales adaptaciones.

- Selección: Es la adaptación por selección de una entre varias alternativas pre-configuradas.
- Habilidad: Consiste en la adaptación por activación o desactivación de componentes.
- Reconfiguración: Modificación de la interfaz de usuario usando componentes predefinidos.

- Edición: Es la adaptación sin restricciones, usando un lenguaje de programación/configuración específico.

6.2.6.2 REQUERIMIENTOS FUNCIONALES

1. Registrar usuario
2. Validar usuario
3. Generar listado de Compañías
4. Generar listado de Proyectos
5. Generar listado de DNIS
6. Generar listado de Resources
7. Calcular número de llamadas entrantes
8. Calcular total de llamadas.
9. Generar grafica
10. Generar zoom

Tabla 4. Requerimiento funcional registrar usuario

IDENTIFICADOR	R-1	NOMBRE	Registrar usuario
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA	Nombre, Cedula, Tipo usuario: (Administrador, operador, cliente)	SALIDA	Usuario nuevo registrado.
DESCRIPCION			
<p>Precondición: Datos debidamente diligenciados Solo el personal autorizado (administrador) puede registrar al usuario Descripción: Notificación de usuario nuevo o agregar usuario que interactúa con el sistema Escribir nombre e ID Seleccionar tipo de usuario Registrar usuario Post condición: 1. Guarda registro de usuario en el sistema</p>			
MANEJO DE SITUACIONES ANORMALES			
<p>No selecciona tipo de usuario Error: Debe seleccionar tipo de usuario para especificar los permisos de este en el sistema.</p>			
CRITERIOS DE ACEPTACIÓN			
<p>El registro del usuario es almacenado exitosamente Mostrar mensaje de error por dato no diligenciado</p>			

Fuente. El autor

Tabla 5. Requerimiento funcional validar usuario

IDENTIFICADOR	R-2	NOMBRE	Validar usuario
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	R1
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA	Nombre, Cedula, Tipo usuario: (Administrador, operador, cliente)	SALIDA	Usuario validado.
DESCRIPCION			
<p>Precondición: Datos debidamente diligenciados Solo el personal autorizado (administrador) puede validar el usuario Descripción: Escribir nombre e ID Seleccionar tipo de usuario Notificación de usuario registrado Post condición: 1. Mensaje de usuario registrado</p>			
MANEJO DE SITUACIONES ANORMALES			
<p>No selecciona tipo de usuario Error: Debe seleccionar tipo de usuario para especificar los permisos de este en el sistema.</p>			
CRITERIOS DE ACEPTACIÓN			
<p>El registro del usuario es almacenado exitosamente Mostrar mensaje de error por dato no diligenciado correctamente</p>			

Fuente. El autor

Tabla 6. Requerimiento funcional generar listado de compañías

IDENTIFICADOR	R-3	NOMBRE	Generar listado de compañías
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA	Compañías	SALIDA	Lista de compañías
DESCRIPCION			
Precondición: Inicializado por el operador o el administrador Cumplir datos de entrada Descripción: Muestra en un listado las compañías Seleccionar compañía Post condición: Generación de una lista despegable			
MANEJO DE SITUACIONES ANORMALES			
Datos de entrada no encontrados Error: Los datos solicitados están incompletos			
CRITERIOS DE ACEPTACIÓN			
La generación de compañías es satisfactoria Muestra mensaje de error de datos no encontrados			

Fuente. El autor

Tabla 7. Requerimiento funcional generar listado de proyectos

IDENTIFICADOR	R-4	NOMBRE	Generar listado de proyectos
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA	Proyectos	SALIDA	Lista de proyectos
DESCRIPCION			
<p>Precondición: Inicializado por el operador o el administrador Cumplir datos de entrada Descripción: Muestra en un listado los proyectos Seleccionar proyecto Post condición: Generación de una lista despegable con los proyectos y su código asociado</p>			
MANEJO DE SITUACIONES ANORMALES			
<p>Datos de entrada no encontrados Error: Los datos solicitados están incompletos</p>			
CRITERIOS DE ACEPTACIÓN			
<p>La generación de la lista de proyectos es satisfactoria Muestra mensaje de error de datos no encontrados</p>			

Fuente. El autor

Tabla 8. Requerimiento funcional generar listado de DNIS

IDENTIFICADOR	R-5	NOMBRE	Generar listado de DNIS
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA	DNIS	SALIDA	Lista de DNIS
DESCRIPCION			
<p>Precondición: Inicializado por el operador o el administrador Cumplir datos de entrada Descripción: Muestra en un listado los DNIS Seleccionar DNI Post condición: Generación de una lista despegable con los DNIS y su código asociado</p>			
MANEJO DE SITUACIONES ANORMALES			
<p>Datos de entrada no encontrados Error: Los datos solicitados están incompletos</p>			
CRITERIOS DE ACEPTACIÓN			
<p>La generación de la lista de DNIS es satisfactoria Muestra mensaje de error de datos no encontrados</p>			

Fuente. El autor

Tabla 9. Requerimiento funcional generar listado de resources

IDENTIFICADOR	R-6	NOMBRE	Generar listado de Resources
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA	Resource	SALIDA	Lista de Resources
DESCRIPCION			
<p>Precondición: Inicializado por el operador o el administrador Cumplir datos de entrada Descripción: Muestra en un listado los Resources Seleccionar Resource Post condición: Generación de una lista despegable con los Resources y su código asociado</p>			
MANEJO DE SITUACIONES ANORMALES			
<p>Datos de entrada no encontrados Error: Los datos solicitados están incompletos</p>			
CRITERIOS DE ACEPTACIÓN			
<p>La generación de la lista de Resources es satisfactoria Muestra mensaje de error de datos no encontrados</p>			

Fuente. El autor

Tabla 10. Requerimiento funcional calcular numero de llamadas entrantes

IDENTIFICADOR	R-7	NOMBRE	Calcular número de llamadas entrantes
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA	Fecha	SALIDA	Cuadro con el número de llamadas entrantes
DESCRIPCION			
<p>Precondición:</p> <ol style="list-style-type: none"> 1. Seleccionar fecha en el calendario <p>Descripción:</p> <ol style="list-style-type: none"> 1. Se hace la consulta de llamadas en el día seleccionado 2. La fórmula accede a los datos correspondientes <p>Post condición:</p> <ol style="list-style-type: none"> 1. Se muestra en un cuadro el número total de llamadas entrantes 			
MANEJO DE SITUACIONES ANORMALES			
<ol style="list-style-type: none"> 1. No se encuentran datos del día seleccionado <p>Error: No es posible calcular el número de llamadas debido a falta de datos de entrada.</p>			
CRITERIOS DE ACEPTACIÓN			
<p>Se muestra el número de llamadas en un cuadro satisfactoriamente</p> <p>Muestra mensaje de error por datos faltantes en los campos</p>			

Fuente. El autor

Tabla 11. Requerimiento funcional calcular total llamadas

IDENTIFICADOR	R-8	NOMBRE	Calcular total llamadas
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA	Fecha	SALIDA	Cuadro con el número total de llamadas entrantes
DESCRIPCION			
<p>Precondición: Seleccionar fecha en el calendario Descripción: Se hace la consulta de llamadas en el día seleccionado La fórmula accede a los datos correspondientes Post condición: Se muestra en un cuadro el número total de llamadas</p>			
MANEJO DE SITUACIONES ANORMALES			
<p>No se encuentran datos del día seleccionado Error: No es posible calcular el número de llamadas debido a falta de datos de entrada.</p>			
CRITERIOS DE ACEPTACIÓN			
<p>Se muestra el número de llamadas en un cuadro satisfactoriamente Muestra mensaje de error por datos faltantes en los campos</p>			

Fuente. El autor

Tabla 12. Requerimiento funcional generar grafica

IDENTIFICADO R	R-9	NOMBRE	Generar grafica
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA		SALIDA	
DESCRIPCION			
<p>Precondición: Campos llenos debidamente diligenciados</p> <p>Descripción: Se muestra la gráfica generada por el día seleccionado del calendario Se muestra el número de llamadas simultaneas</p> <p>Post condición: 1. Se muestra la gráfica en el plano</p>			
MANEJO DE SITUACIONES ANORMALES			
<p>La grafica se presenta de manera anormal e ilegible</p> <p>Error: La fecha no se muestra</p>			
CRITERIOS DE ACEPTACIÓN			
<p>La grafica se muestra de una forma normal y amigable totalmente legible</p> <p>Se muestra el cuadro de llamadas entrantes y llamadas totales analizadas</p>			

Fuente. El autor

Tabla 13. Requerimiento funcional generar zoom

IDENTIFICADOR	R-10	NOMBRE	Generar zoom
TIPO	Necesario	REQ. QUE LO UTILIZA O ESPECIALIZA	_____
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	_____
ENTRADA		SALIDA	
DESCRIPCION			
<p>Precondición: Grafica generada previamente en el plano Descripción: Se muestra la gráfica generada por el día seleccionado del calendario Se hacer un zoom en los puntos generados Post condición: 1. Se muestra el zoom de la gráfica en el plano</p>			
MANEJO DE SITUACIONES ANORMALES			
<p>La grafica se presenta de manera anormal e ilegible Error: La grafica generada se distorsiona y no es entendible</p>			
CRITERIOS DE ACEPTACIÓN			
<p>La grafica se muestra de una forma normal y amigable totalmente legible Se muestra el cuadro de llamadas entrantes y llamadas totales analizadas</p>			

Fuente. El autor

6.2.6.2 REQUERIMIENTOS NO FUNCIONALES

Atributos de calidad

1. La aplicación debe ser web.
2. La aplicación debe estar disponible las 24 horas del día.

Documentación requerimientos no funcionales

Tabla 14. Requerimiento no funcional aplicación web

IDENTIFICADOR	RN-1	NOMBRE	Aplicación web
TIPO	Necesario	REQ. FUNCIONAL ASOCIADO	Todos
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	Todos
DESCRIPCION			
Precondición: 1. Hardware y software disponible 2. Conexión a internet Descripción: 1. Se establece los requerimientos que el usuario necesita 2. Se propone el software para desarrollo de la aplicación web Post condición: 1. Lenguaje HTML y java 2. Primefaces Framework			
MANEJO DE SITUACIONES ANORMALES			
1. No se encuentra conectado a internet 2. Error: Muestra mensaje falla conexión a internet			
CRITERIOS DE ACEPTACIÓN			
1. Muestra la aplicación funcionando con conexión a internet 2. Muestra mensaje de error de conexión			

Fuente. El autor

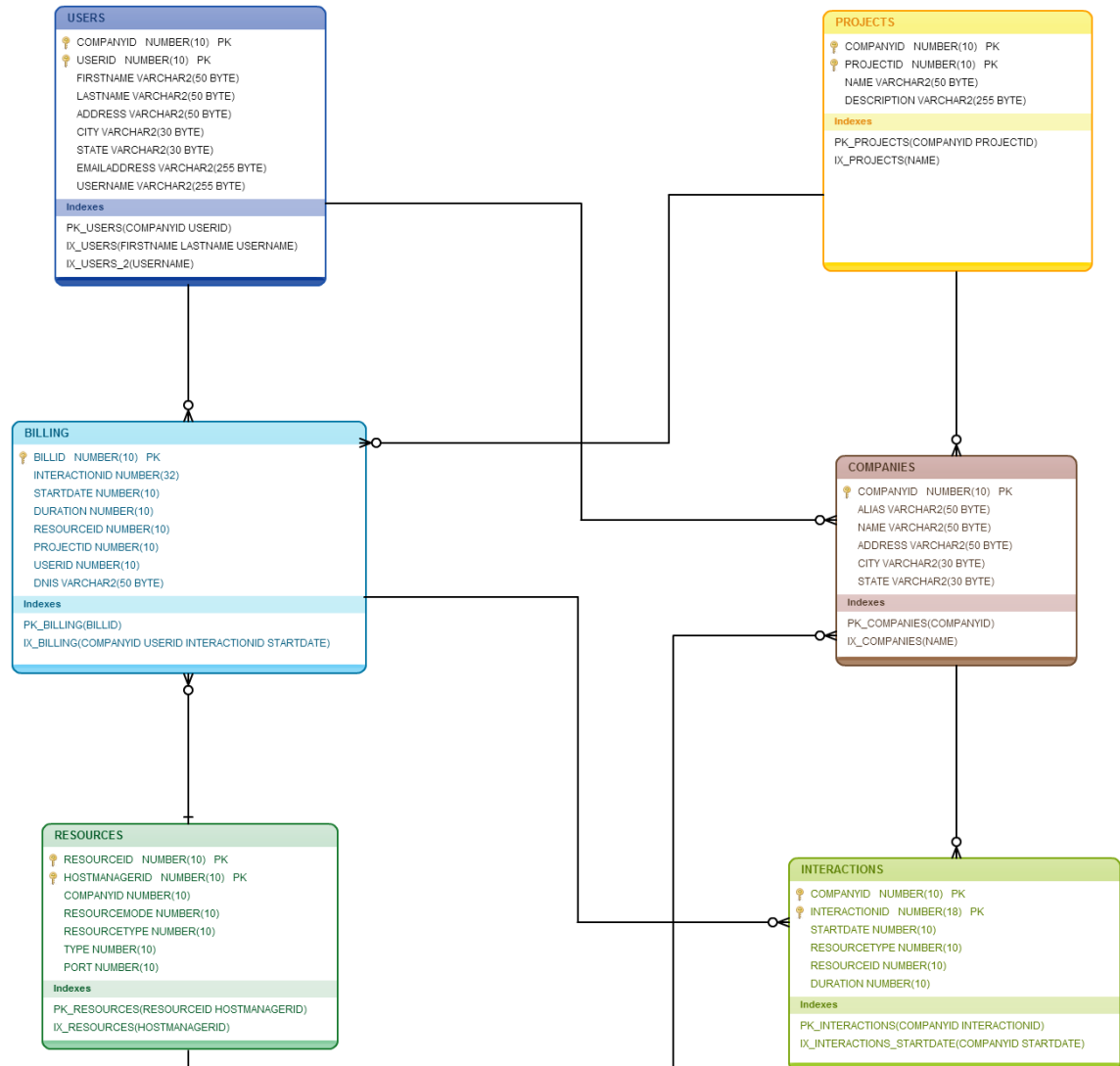
Tabla 15. Requerimiento no funcional aplicación disponible

IDENTIFICADOR	RN-2	NOMBRE	Aplicación disponible
TIPO	Necesario	REQ. FUNCIONAL ASOCIADO	Todos
PRIORIDAD DE DESARROLLO	Alta	DOC. ASOCIADOS	Todos
DESCRIPCION			
Precondición: 1. Hardware y software disponible 2. Conexión a internet Descripción: 1. Se controla que el servidor de aplicaciones esté disponible las 24 horas Post condición: 1. La aplicación se muestra disponible			
MANEJO DE SITUACIONES ANORMALES			
1. No se encuentra conectado a internet 2. Error: Muestra mensaje falla conexión a internet			
CRITERIOS DE ACEPTACIÓN			
1. Muestra la aplicación funcionando con conexión a internet 2. La grafica se muestra con los componentes correspondientes			

Fuente. El autor

ANEXO F: DIAGRAMA DE BD/MODELO RELACIONAL

Figura 12. Diagrama modelo base da datos



Fuente. El autor

El modelo relacional de la base de datos comprende 6 tablas cada una de ellas interrelacionadas y presentadas únicamente como un modelo similar al utilizado en la empresa de Contact Center.

A continuación se explican las tablas utilizadas en la implementación del proyecto:

Tabla 16. Descripción de tablas modelo relacional

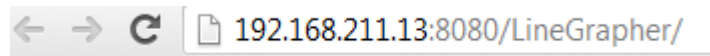
Nombre	Descripción
Billing	Tabla principal donde se asocian las compañías los proyectos y los usuarios, es responsable de la facturación de las interacciones.
Companies	Se encuentran todas las compañías asociadas a la empresa cada una con un identificador que la diferencia de las demás.
Projects	Comprende todos los proyectos que tiene la empresa y dentro de ellos los grupos de trabajo asociados de cada campaña.
Interactions	Presenta las interacciones entre los usuarios de la compañía estas determinadas por una fecha de inicio y una duración.
Users	Contiene los usuarios de las campañas cada usuario tiene un identificador que lo asocia a la compañía a la que pertenece.
Resources	Recursos consumidos por las campañas

El autor


ANEXO G: MANUAL DE USUARIO

Este manual esta hecho para usuarios finales los cuales son personas del área de tecnología que están dentro del contexto de los términos mencionados y listas desplegadas por la aplicación.

1. Como primera medida ingresamos a la siguiente dirección donde esta la aplicación desde el browser que tengamos instalado en nuestro equipo proporcionándole la siguiente dirección:

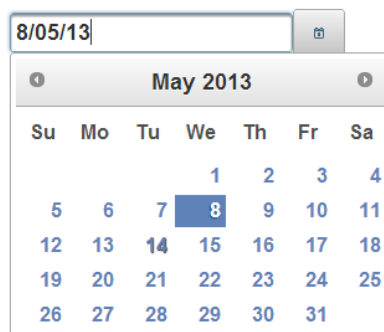


2. Para ingresar a la aplicación es necesario tener un usuario y contraseña esto con el fin de que la aplicación sea segura y no sea accedida por usuarios no autorizados, al iniciar el sistema se muestra el siguiente formulario en el cual se debe ingresar el nombre de usuario y clave de acceso al sistema

A screenshot of a login form titled 'Ingreso'. It contains two input fields: 'Usuario:' with the value 'admin' and 'Clave:' with masked characters '.....'. Below the fields is a button labeled 'Ingreso'.


Dependiendo de los roles que hayan sido asignados se podrá acceder a cada uno de los módulos del sistema

3. Como parámetro de entrada se tiene que seleccionar un día en el calendario haciendo click en el




4. Seguido de esto dependiendo de la gráfica que se quiera generar se determinan los siguientes criterios como: Compañía, Proyecto, DNIS o Resource haciendo click en cada uno de ellos

8/05/13 

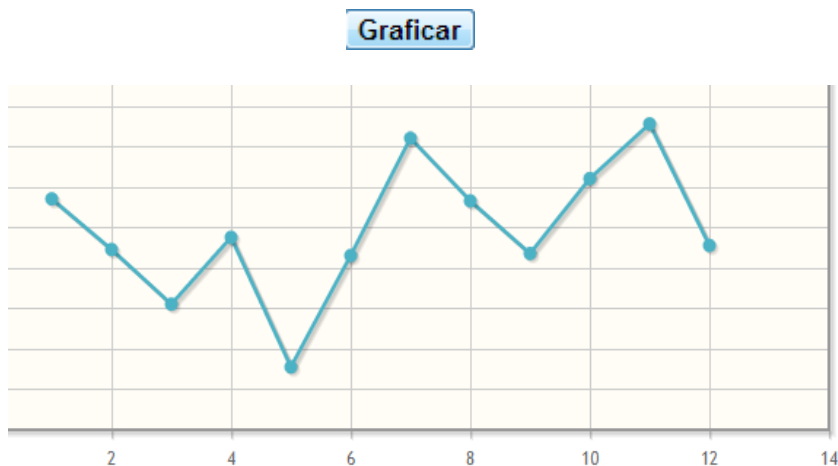
FONTIBON 

Prueba 



3537333 

15 

5. Por último se da click al botón graficar donde se genera la gráfica correspondiente de las llamadas simultáneas de los criterios definidos anteriormte.



6. Se visualiza en un cuadro la información consultada

	Numero de llamadas simultaneas = 125
	Total de llamadas = 1535

7. Si queremos exportar un reporte de las llamas simulaneas damos click en el botón exportar reporte dependiendo del formato en que se quiera visualizar:

Exportar reporte en Excel

Exportar reporte en pdf

ANEXO H: MANUAL DEL SISTEMA

Este manual está dirigido al personal del área de tecnología responsable de subir y configurar finalmente la aplicación en el servidor de aplicaciones; por tanto se asume que el lector está familiarizado con conceptos básicos de administración en Windows y, para la puesta en práctica de los conocimientos explicados aquí cuenta con un usuario que tiene los permisos adecuados (normalmente un usuario Administrador de Windows).

Igualmente debe estar familiarizado con la funcionalidad de la aplicación, desde el punto de vista de usuario, la aplicación web desarrollada está diseñada para soportar portabilidad en su plataforma de Base de Datos Oracle 11g.

Características Generales

Sistema Operativo: Windows 7

Lenguaje de Desarrollo: Java 1.7.0_15

Espacio en disco: 247 MB

Memoria Ram minima requerida: 512 MB

Windows 7 Ultimate

Copyright © 2009 Microsoft Corporation. Reservados todos los derechos.

Service Pack 1

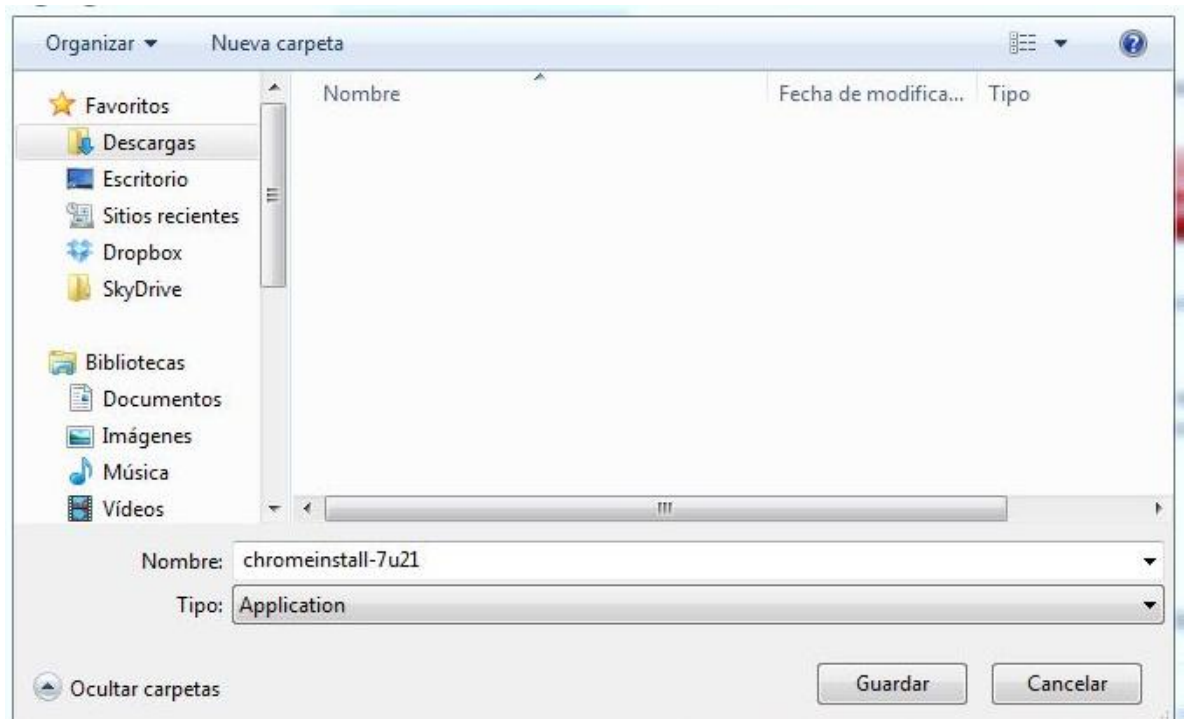


Instalacion Java

1. Verificar si existe una conexión a internet
2. Ir a la siguiente dirección www.java.com/es/download/ para descargar la version de java requerida
3. Oprimir el botón Descarga gratuita de Java



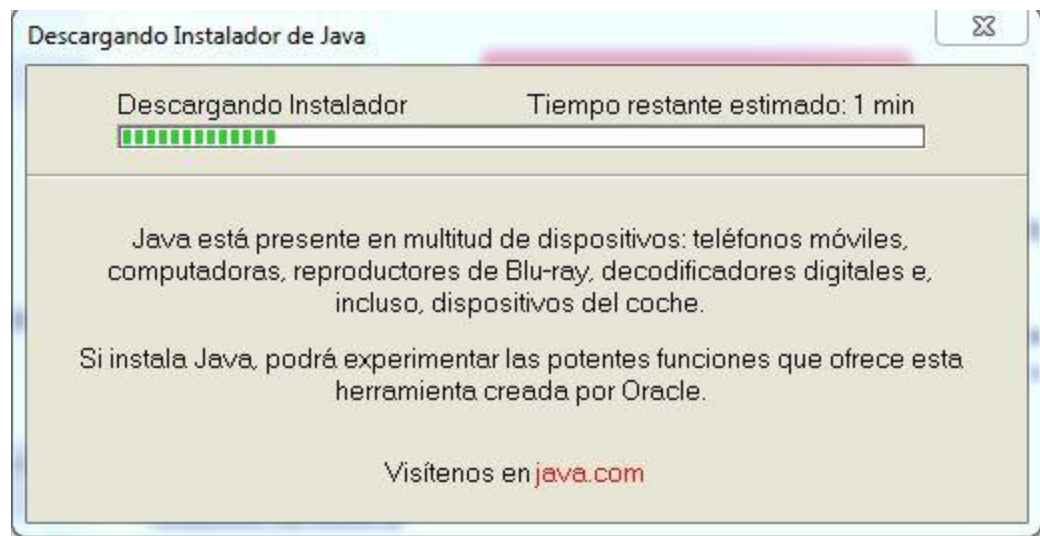
4. Escoger una ruta donde se va a dejar el archivo



5. Al ejecutar el archivo descargado se mostrará una ventana como la siguiente, damos click en “Instalar” y éste nos asignara una ruta por defecto



6. El instalador necesitará de una conexión a internet para realizar la descarga



7. Si proceso fue exitoso veremos una ventana como la siguiente:



Conexión a la base de datos

1. Para la conexión a la base de datos necesitamos el driver ojdbc14.jar de Oracle lo podemos descargar de la siguiente dirección

<http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-10201-088211.html>

2. Damos click en Accept license Agreement para aceptar la licencia

Oracle Database 10g Release 2 JDBC Drivers

You must accept the OTN Development and Distribution License Agreement to download this software.

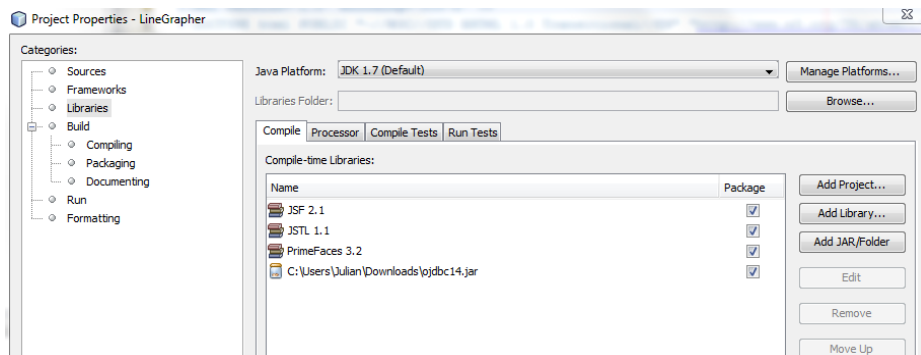
☒ Accept License Agreement | ☐ Decline License Agreement

3. Seguido de esto procedemos a descargarlo

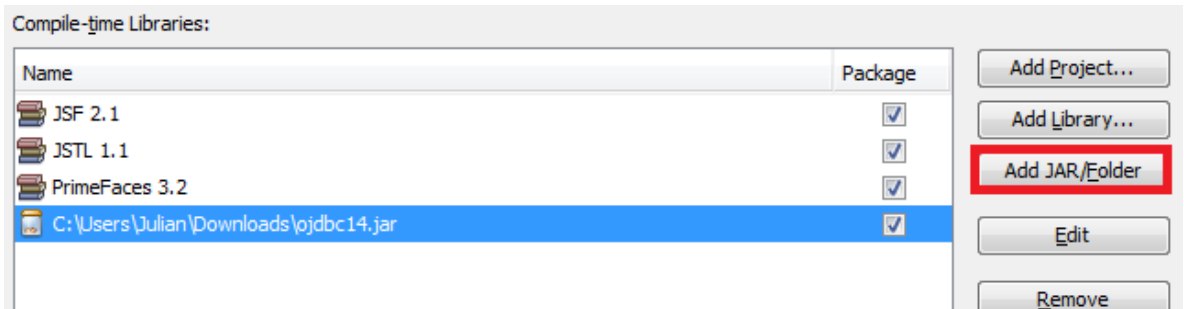
JDBC Thin for All Platforms

- ☐ classes12.jar (1,621,582 bytes) - for use with JDK 1.2 and 1.3
- ☐ classes12_g.jar (2,070,298 bytes) - same as classes12.jar, but compiled with "javac -g" and contain some tracing information
- ☐ classes12dms.jar (1,629,454 bytes) - same as classes12.jar, but contain additional code to support Oracle Dynamic Monitoring Services
- ☐ classes12dms_g.jar (2,078,718 bytes) - same as classes12dms.jar, but compiled with "javac -g" and contain some tracing information
- ☐ ojdbc14.jar (1,569,316 bytes) - classes for use with JDK 1.4 and 1.5
- ☒ ojdbc14_g.jar (1,965,422 bytes) - same as ojdbc14.jar, but compiled with "javac -g" and contain some tracing information
- ☐ ojdbc14dms.jar (1,576,882 bytes) - same as ojdbc14.jar, but contain additional code to support Oracle Dynamic Monitoring Services
- ☐ ojdbc14dms_g.jar (1,973,634 bytes) - same as ojdbc14dms.jar, but were compiled with "javac -g" and contain some tracing information
- ☐ demo.tar (604,160 bytes) - contains sample JDBC programs

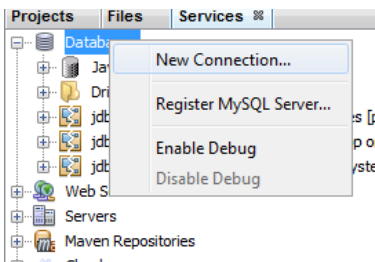
4. Una vez descargado lo agregamos a nuestro proyecto dando click con el segundo botón en las propiedades de nuestra aplicación donde visualizamos la siguiente pantalla



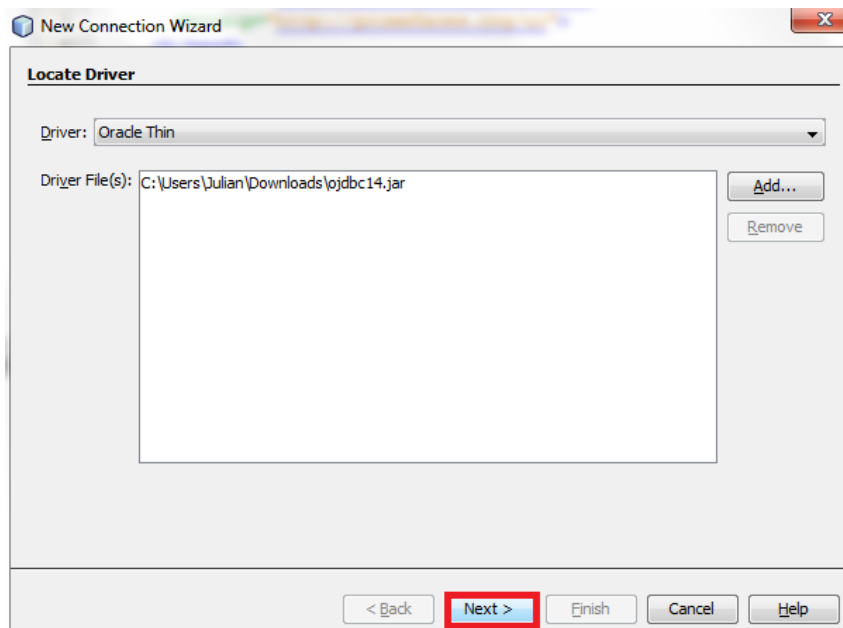
5. Damos click en Add Jar



6. Seguido de esto vamos a la parte donde se encuentra la base de datos para crear la conexión



7. Seleccionamos Oracle Thin y agregamos el .jar para crear la conexión a la base de datos, damos clien en next



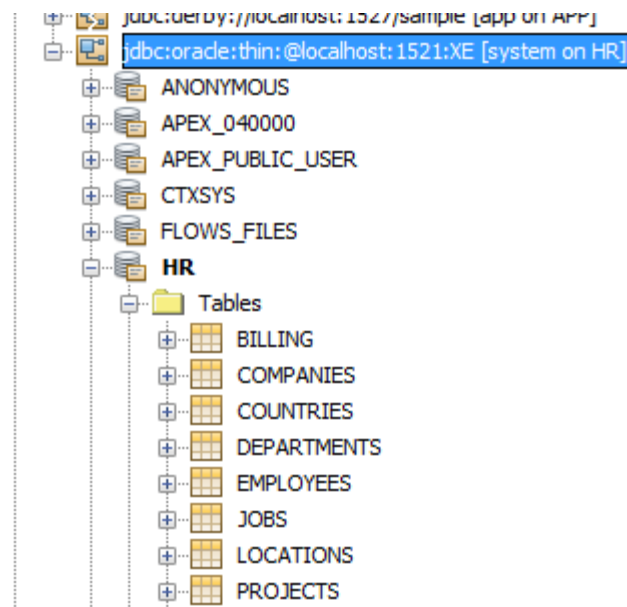
8. Se presentara una pantalla donde se proporcionara el usuario y la contraseña de la case de datos asi como también la dirección(JDBC URL)

The screenshot shows the 'New Connection Wizard' dialog box, specifically the 'Customize Connection' step. The fields are as follows:

- Driver Name: Oracle Thin (Service ID (SID))
- Host: localhost
- Port: 1521
- Service ID (SID): XE
- User Name: hr
- Password: (masked with dots)
- ☐ Remember password
- Test Connection button
- JDBC URL: jdbc:oracle:thin:@localhost:1521:XE

At the bottom, there are navigation buttons: '< Back', 'Next >' (highlighted with a red box), 'Finish', 'Cancel', and 'Help'.

9. Seguido de esto se visualiza la conexión a la base de datos con sus tablas correspondientes



Instalacion Primefaces

1. Para la instalación de Primefaces nos vamos a la siguiente dirección

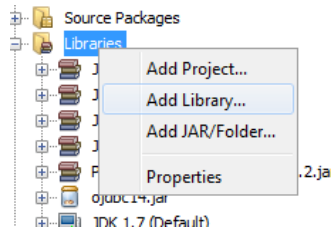
<http://www.primefaces.org/downloads.html>

2. Descargamos primefaces-3.4.2 jar

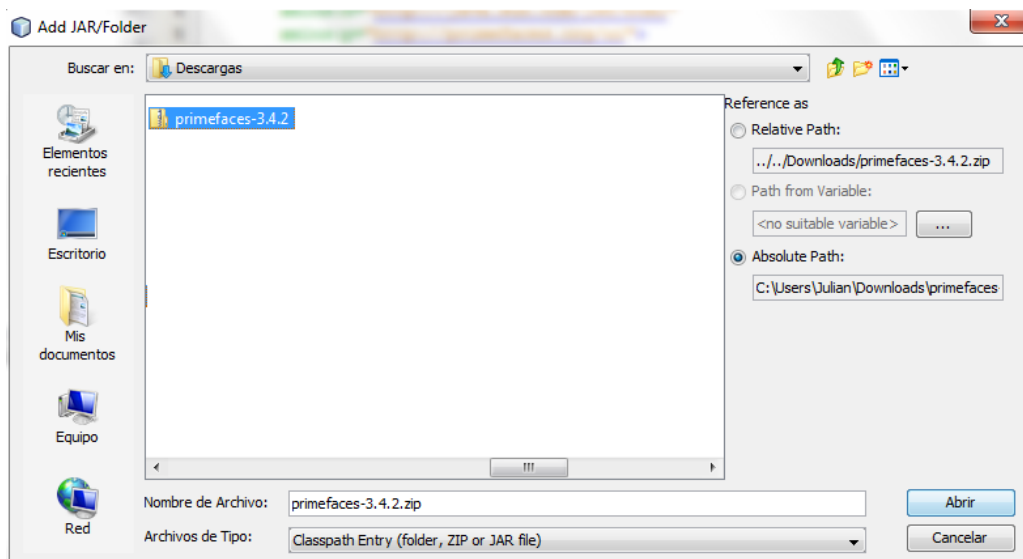
Community Downloads

Version	Binary
3.5	primefaces-3.5.jar
3.4.2	primefaces-3.4.2.jar
3.4.1	primefaces-3.4.1.jar
3.4	primefaces-3.4.jar

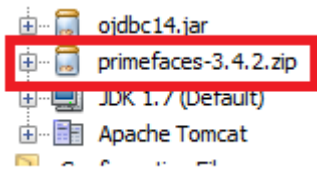
3. Lo agregamos en las librerías a nuestro proyecto



4. Damos click en abrir



5. Finalmente verificamos que este en nuestro proyecto

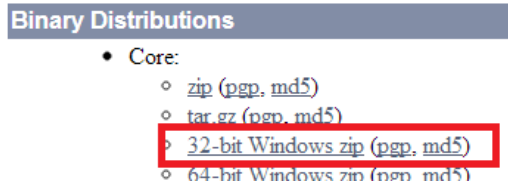


Instalacion Tomcat 7.0.39

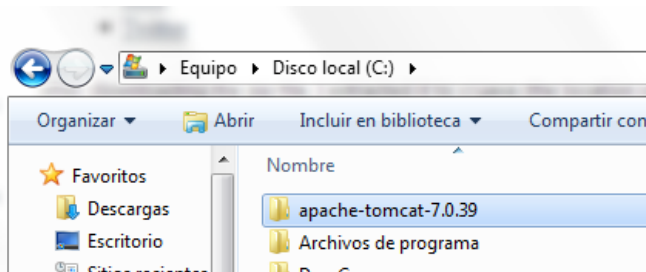
1. Para la instalación de Tomcat es necesario dirigirnos a la siguiente dirección:

<http://tomcat.apache.org/download-70.cgi>

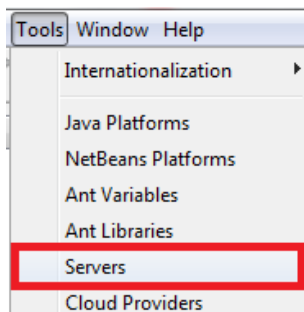
2. Descargamos la version 7.0.39 de 32 bits



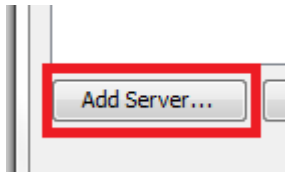
3. Una vez descargada descomprimos la carpeta y la dejamos en el disco local C:



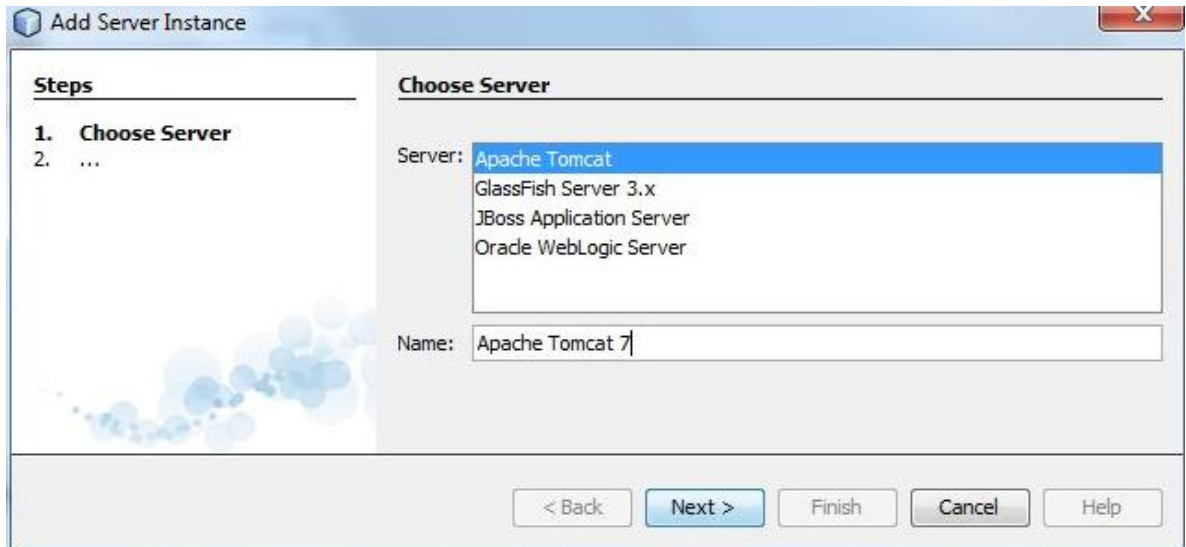
4. Seguido de esto vamos a nuestro IDE donde están los servidores



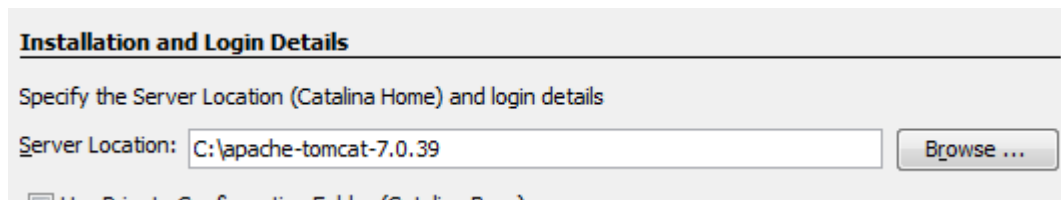
5. Damos click en add server



6. Seleccionamos nuestro servidor en este caso Apache Tomcat



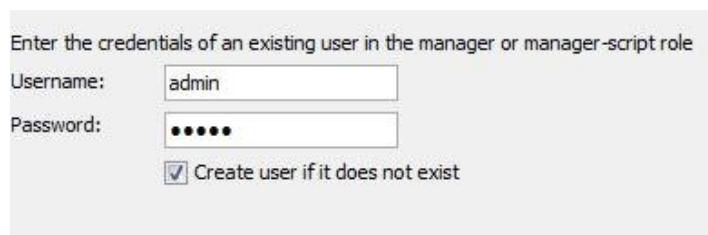
7. Seleccionamos donde esta la ruta de nuestro servidor la cual definimos anteriormente



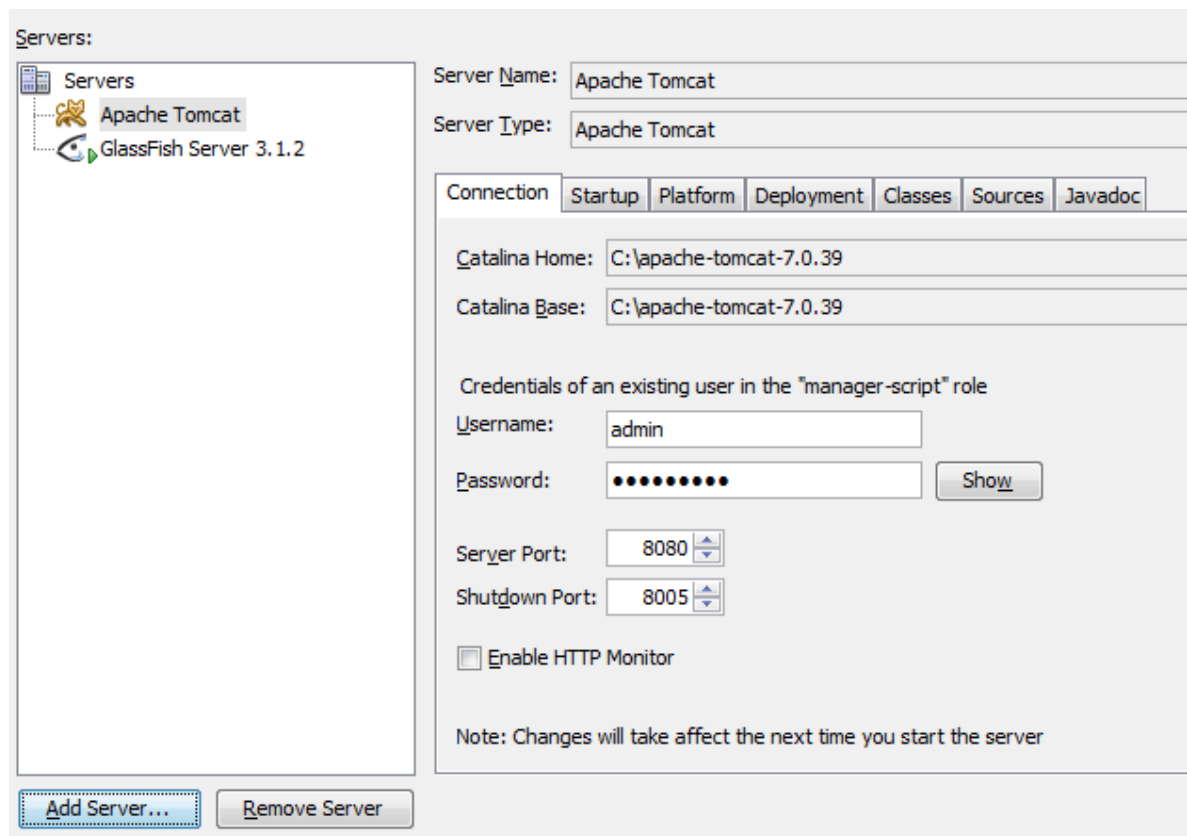
8. Proporcionamos las siguientes credenciales

Username: admin

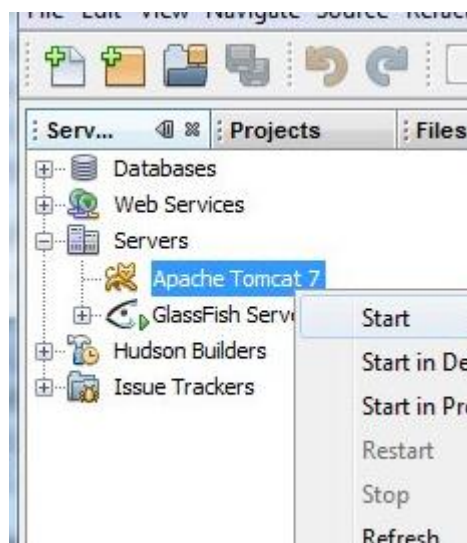
Password: admin

A screenshot of a form for entering credentials. It has a title 'Enter the credentials of an existing user in the manager or manager-script role'. There are two input fields: 'Username:' with 'admin' entered, and 'Password:' with 'admin' entered (represented by dots). Below these fields is a checkbox labeled 'Create user if it does not exist' which is checked.

9. Visualizamos la siguiente pantalla la cual debemos dejar por defecto la siguiente configuración como se muestra en la imagen:



10. Finalmente vemos nuestro servidor damos click en Start



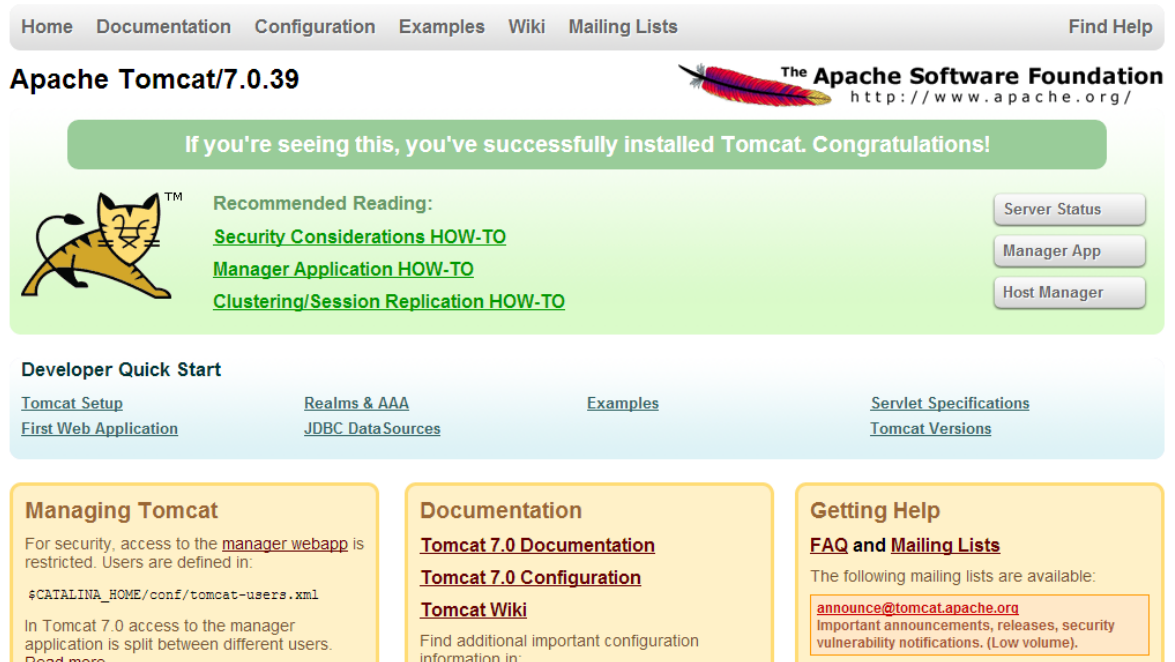
11. Visualizamos en la consola que no exitan errores

```
Jan 2, 2012 6:58:53 AM org.apache.catalina.core.ApplicationContext log
INFO: ContextListener: contextInitialized()
Jan 2, 2012 6:58:53 AM org.apache.catalina.core.ApplicationContext log
INFO: SessionListener: contextInitialized()
Jan 2, 2012 6:58:53 AM org.apache.catalina.core.ApplicationContext log
INFO: ContextListener: attributeAdded('org.apache.jasper.compiler.TldLocationsCache', 'org
```

12. Para probar que el servidor este arriba vamos a la siguiente dirección:

<http://localhost:8080/>

Donde nos tiene que salir la siguiente pantalla




Home Documentation Configuration Examples Wiki Mailing Lists Find Help

Apache Tomcat/7.0.39

The Apache Software Foundation <http://www.apache.org/>

If you're seeing this, you've successfully installed Tomcat. Congratulations!

 Recommended Reading:

- [Security Considerations HOW-TO](#)
- [Manager Application HOW-TO](#)
- [Clustering/Session Replication HOW-TO](#)

Server Status
Manager App
Host Manager

Developer Quick Start

- [Tomcat Setup](#)
- [Realms & AAA](#)
- [Examples](#)
- [Servlet Specifications](#)
- [First Web Application](#)
- [JDBC DataSources](#)
- [Tomcat Versions](#)

Managing Tomcat

For security, access to the [manager webapp](#) is restricted. Users are defined in:

```
$CATALINA_HOME/conf/tomcat-users.xml
```

In Tomcat 7.0 access to the manager application is split between different users. [Read more](#)

Documentation

- [Tomcat 7.0 Documentation](#)
- [Tomcat 7.0 Configuration](#)
- [Tomcat Wiki](#)

Find additional important configuration information in:

Getting Help

FAQ and Mailing Lists

The following mailing lists are available:

- announce@tomcat.apache.org
Important announcements, releases, security vulnerability notifications. (Low volume).

Se describe a continuación los pasos para subir la aplicación web al servidor de aplicaciones:

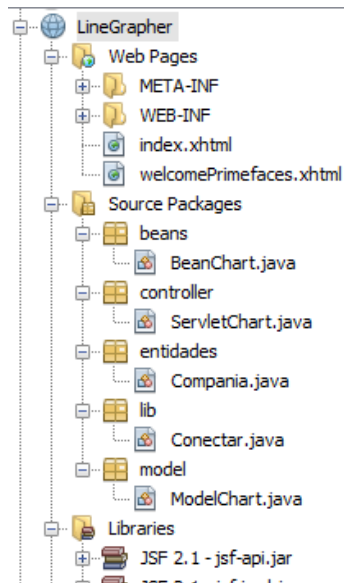
1. Generar un archivo .War

La aplicación web es empaquetada o comprimida lista para ser desplegada en cualquiera de los servidores que brindan servicios para este tipo de aplicaciones web.

La gran ventaja de los WAR es que facilitan los detalles de configuración, siempre que un entorno de programación lo automatice.

Estructura de un archivo war

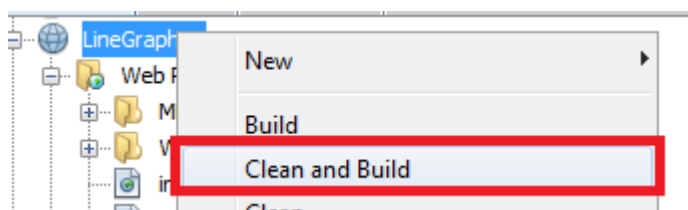
La estructura de un archivo war es la misma que usa la aplicación web. En la figura podemos ver la estructura de directorios.



Formas de crear un archivo war

- Usando un IDE
 - Después de haber creado la aplicación en el IDE (Entorno de Desarrollo Integrado) nos genera el archivo .war automáticamente.
 - Para entender de mejor manera la creación del archivo war de una aplicación web usamos como IDE Netbeans 7.3

2. Hacer clic derecho sobre el nombre de la aplicación web (LineGrapher) y luego Clean and Build.



Esto genera una carpeta que contiene el archivo war de la aplicación

3. Usando la consola de comandos de Windows 7

- Buscar el directorio raíz de la aplicación mediante la consola de comandos de Windows

```
C:\Archivos de programa\Apache Software Foundation\Apache Tomcat 6.0.14\webapps\
aplicacionweb>
```

4. Luego de esto tecleamos el siguiente comando:

jar cv0f LineGrapher.war .

5. Podemos observar el punto obligatorio que hay al final de la línea; le dice al programa jar que archive el directorio actual

7. CONCLUSIONES

Por medio de la ingeniería de sistemas se ha contribuido a la realización de un proyecto para la empresa de Contact Center, y al desarrollo tecnológico en la empresa, ofreciendo una nueva forma de presentar la información.

La aplicación Web benefició al área de tecnología para acceder a la información de una forma más amigable, ajustándose a las necesidades que se plantearon.

El desarrollo de esta aplicación fue posible a través del seguimiento de una metodología adecuada, donde se plantearon 3 fases: Inicial, Desarrollo e Integración.

Para aceptar el proyecto, se realizaron varias pruebas por el personal encargado de soluciones informáticas.

Realizar un buen diseño y una buena planeación da garantías de que la implementación tenga mayores probabilidades de ejecutarse óptima y eficientemente

8. RECOMENDACIONES

La empresa de Contact Center será la encargada del manejo de la aplicación web, en el momento de realizar modificaciones por parte del área de soluciones informáticas.

De igual manera será responsable de informar el funcionamiento de la aplicación con los criterios establecidos que se deben tener en cuenta para generar las gráficas, una vez puesta esta en marcha y aprobada por el área.

BIBLIOGRAFIA

COULOURIS, George. Sistemas Distribuidos: Conceptos y Diseño. Addison-Wesley. Madrid: 2001. 744 p. [Citado Mayo 1 de 2013]

CORE JAVASERVER FACES, D. Geary and C. Horstmann.. Prentice hall, 3 ed. (2010) [Citado Mayo 8 de 2013]

GNU Operating System. Disponible en <http://www.gnu.org/licenses/licenses.es.html> [Citado Mayo 17 de 2013]

INSTITUTO COLOMBIANO DE NORMAS TÉCNICAS Y CERTIFICACIÓN. Documentación: presentación de Tesis, trabajos de grado y otros trabajos de investigación. Bogotá D.C.:2005. 112 p. [Citado Mayo 13 de 2013]

MCGRAW-HIL (2002). Ingeniería de Software Un enfoque práctico [Citado Mayo 15 de 2013]

PRIMEFACES Disponible en <http://www.primefaces.org/> [Citado Mayo 17 de 2013]

THE APACHE SOFTWARE FOUNDATION. Disponible en <http://www.apache.org/>. [Citado Mayo 7 de 2013]